

**ВИЩИЙ НАВЧАЛЬНИЙ ЗАКЛАД УКООПСІЛКИ
«ПОЛТАВСЬКИЙ УНІВЕРСИТЕТ ЕКОНОМІКИ І
ТОРГІВЛІ»**

**ІНСТИТУТ ЕКОНОМІКИ, УПРАВЛІННЯ ТА
ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ
ФАКУЛЬТЕТ ЕКОНОМІКИ І МЕНЕДЖМЕНТУ
ФОРМА НАВЧАННЯ ДЕННА
КАФЕДРА МАТЕМАТИЧНОГО МОДЕЛЮВАННЯ ТА
СОЦІАЛЬНОЇ ІНФОРМАТИКИ**

Допускається до захисту
Завідувач кафедри _____ О.О. Ємець
(підпис)

«_____» _____ 2019 р.

**ПОЯСНЮВАЛЬНА ЗАПИСКА
ДО ДИПЛОМНОЇ РОБОТИ
на тему
ТРЕНАЖЕР З ТЕМИ "ЛОГІКА ВИСЛОВЛЮВАНЬ" ДИСТАНЦІЙНОГО
НАВЧАЛЬНОГО КУРСУ "МАТЕМАТИЧНА ЛОГІКА" ТА РОЗРОБКА
ЙОГО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ
зі спеціальності 122 «Комп'ютерні науки»**

Виконавець роботи Сивокінь Олександр Юрійович
_____ «18» листопада 2019р.
(підпис)

Науковий керівник к.ф.-м.н., доц., Черненко Оксана Олексіївна
_____ «__» _____ 201_р.
(підпис)

ПОЛТАВА 2019 р.

**Вищий навчальний заклад Укоопспілки
«Полтавський університет економіки і торгівлі»**

ЗАТВЕРДЖУЮ

Зав. кафедри ММСІ

_____ О.О. Ємець

(підпис)

«__» _____ 20__ р.

**Завдання та календарний графік
виконання дипломної роботи**

Студент(ка) спеціальності 122 «Комп'ютерні науки»

Прізвище, ім'я, по батькові Сивокінь Олександр Юрійович

1. Тема «ТРЕНАЖЕР З ТЕМИ "ЛОГІКА ВИСЛОВЛЮВАНЬ" ДИСТАНЦІЙНОГО НАВЧАЛЬНОГО КУРСУ "МАТЕМАТИЧНА ЛОГІКА" ТА РОЗРОБКА ЙОГО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ» затверджена наказом ректора №1 ____-Н від «02» вересня 2018 р.

Термін подання студентом дипломної роботи «18» листопада 2019 р.

2. Вихідні дані до дипломної роботи курс лекцій «Математична логіка», інформаційні джерела.

3. Зміст пояснювальної записки (перелік питань, які потрібно розробити)

Вступ

1. ПОСТАНОВКА ЗАДАЧІ

1.1. Змістовна (вербальна, словесна) постановка

1.2. Математична (інформаційна) модель та її характеристики

2. ІНФОРМАЦІЙНИЙ ОГЛЯД

2.1. Огляд робіт, де розглянуте аналогічне до теми роботи завдання.

2.2. Позитивні аспекти оглянутих робіт.

2.3. Вади розробок з оглянутих робіт.

2.4. Необхідність та актуальність теми роботи (як наслідок інформаційного огляду).

3. ТЕОРЕТИЧНА ЧАСТИНА

3.1. Алгоритмізація задачі за темою роботи

3.2. Розробка блок-схеми (структурної схеми), яка підлягає програмуванню.

3.3. Обґрунтування вибору програмних засобів для реалізації завдання роботи.

4. ПРАКТИЧНА ЧАСТИНА

4.1. Опис процесу програмної реалізації.

4.2. Опис програми.

4.3. Перевірка валідності (тестування всіх процедур). Дослідження можливостей програмної реалізації.

4.4. Необхідна користувачу програми інструкція.

Висновки

4. Перелік графічного матеріалу (з точним визначенням кількості блок-схем, іншого графічного матеріалу) 6 аркушів блок-схем, ілюстрації за необхідністю.

5. Консультанти розділів дипломної роботи

Розділ	Консультант (прізвище, ініціали)	Підпис, дата	
		Завдання видав	Завдання прийняв
Вступ	Черненко О.О.		
1. Постанова задачі	Черненко О.О.		
2. Інформаційний огляд	Черненко О.О.		
3. Теоретична частина	Черненко О.О.		
4. Практична реалізація	Черненко О.О.		

6. Календарний графік виконання дипломної роботи

Зміст роботи	Термін виконання	Фактичне виконання
1. Вступ		
2. Вивчення методичних рекомендацій та стандартів та звіт керівнику		
3. Постановка задачі		
4. Інформаційний огляд джерел бібліотек та інтернету		
5. Теоретична частина		
6. Практична частина		
7. Закінчення оформлення		
8. Доповідь студента на кафедрі		
9. Доробка (за необхідністю), рецензування		

Дата видачі завдання «__» вересня 20__ р.

Студент(ка) _____
(підпис)

Науковий керівник _____
(підпис) (науковий ступінь, вчене звання, П.І.Б.)

Результати захисту дипломної роботи

Дипломна робота оцінена на _____
(балів, оцінка за національною шкалою, оцінка за ECTS)

Протокол засідання ЕК № _____ від «_____» _____ 201__ р.

Секретар ЕК _____

РЕФЕРАТ

Записка: 89 с., 37 рис., 4 додатка (на 11 сторінках), 27 джерел.

Предмет розробки – програмування тренажера для вивчення логіки висловлювань.

Мета роботи — створити тренажер, який буде використовуватись в навчанні студентів логіки висловлювань.

Методи, які були використані для розв’язування задачі – матеріали лекцій курсу «Математична логіка». Робоча програма розроблена в середовищі Android Studio.

Розроблено тренажер для навчання студентів логіки висловлювань, вивчення логічних операторів та розв’язуванню задач..

Здійснена програмна реалізація тренажера на високорівневій мові програмування JAVA.

Ключові слова: ТРЕНАЖЕР, ДИСТАНЦІЙНА ОСВІТА, ЛОГІКА ВИСЛОВЛЮВАНЬ, МАТЕМАТИЧНА ЛОГІКА.

ЗМІСТ

ПЕРЕЛІК СКОРОЧЕНЬ.....	6
ВСТУП.....	7
1. ПОСТАНОВКА ЗАДАЧІ.....	9
1.1. Змістовна постановка.....	9
1.2. Приклад для створення тренажера.....	9
2. ІНФОРМАЦІЙНИЙ ОГЛЯД.....	11
2.1. Огляд робіт, де розглянуте аналогічне до теми роботи завдання... 11	
2.2. Позитивні аспекти оглянутих робіт.....	11
2.3. Вади розробок з оглянутих робіт.....	12
2.4. Необхідність та актуальність теми роботи.....	12
3. ТЕОРЕТИЧНА ЧАСТИНА.....	13
3.1. Алгоритмізація задачі за темою роботи.....	13
3.2. Блок-схема.....	24
3.3. Обґрунтування вибору програмних засобів для реалізації завдання роботи.....	30
4. ПРАКТИЧНА ЧАСТИНА.....	31
4.1. Опис процесу програмної реалізації.....	31
4.2. Опис програми.....	33
4.3. Перевірка валідності.....	35
4.4. Необхідна користувачу програми інструкція.....	44
ВИСНОВКИ.....	45
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	46
ДОДАТКИ.....	51
Додаток А Компакт-диск з тренажером.....	51
Додаток Б.1 Програмний код програми-тренажера (крок 1.1).....	52
Додаток Б.2 Програмний код програми-тренажера (крок 2.1.1).....	58
Додаток Б.3 Програмний код програми-тренажера (крок 3.1.1).....	62

ПЕРЕЛІК СКОРОЧЕНЬ

Умовні позначення, символи, скорочення, терміни	Пояснення умовних позначень, скорочень, символів

ВСТУП

Дистанційна освіта допомагає зменшити навантаження в навчанні як для студента так і для викладача. Вона дозволяє навчатися та отримувати необхідні знання віддалено від навчального закладу в будь-який зручний час та будь-якому місці. Наразі, в ПУЕТ працює система дистанційної освіти. Постійно йде вдосконалення дистанційних курсів, а також розробляються нові. Важливою їх складовою є тренажери. Одним з таких курсів є «Математична логіка», яка дає змогу визначити істинність або хибність математичного твердження, виходячи з первинних припущень, які називаються аксіомами. Логіку також застосовують в інформатиці для побудови комп'ютерних програм і доведення їх коректності. Поняття, методи й засоби логіки покладено в основу сучасних інформаційних технологій [1].

Актуальність теми магістерської роботи полягає в тому, що створюється тренажер, який має бути використаний в навчанні студента логіці висловлювань.

Мета магістерської роботи — створити тренажер, який буде використовуватись в навчанні студентів логіці висловлювань спираючись на курс «Математична логіка».

Об'єкт розробки — матеріальне забезпечення ПУЕТ та інтернет ресурси.

Предмет розробки — програмування тренажера.

Завдання бакалаврської роботи:

1. розробити алгоритм тренажера;
2. на основі розробленого алгоритму запрограмувати тренажер для вивчення логіки висловлювань;
3. застосувати набуті знання та навички в програмуванні та вивченні інших дисциплін;
4. здобути нові знання та навички в програмуванні;

5. впровадити розроблений тренажер в дистанційний курс «Математична логіка».

Методи розробки. При реалізації магістерської роботи було використано середовище розробки Android Studio. При написанні програми використано високорівневу мову Java.

Практичне значення роботи полягає в тому, що даний тренажер можна впроваджувати в дистанційне та стаціонарне навчання студентів для курсу «Математична логіка».

Структура пояснювальної записки до магістерської роботи: вступ, постановка задачі, інформаційний огляд, теоретична частина, практична частина, висновки, список використаних джерел, додатки.

Обсяг пояснювальної записки: 90 стор., в т.ч. основна частина 39 стор., додатки - 11 стор., джерел – 27 назв, рисунків - 37, блок-схем – 6 аркушів.

1. ПОСТАНОВКА ЗАДАЧІ

1.1. Змістовна постановка задачі

В навчанні студента дуже допомагають так звані навчальні тренажери. Вони дають змогу не тільки перевіряти, а й навчати і закріплювати пройдений матеріал.

Програмний продукт має містити наступне: повинно бути реалізовано інтерактивний розв'язок задач (покрокові обчислення та перевірка результатів); проходження, при необхідності, тестових завдань студентом, для індикації теоретичних знань, з допомогою яких розв'язуються поставлені завдання; розв'язання задачі з поясненням і пояснення теорем.

1.2. Приклад для створення тренажера

Тренажер створювався на прикладі завдань з книги «Дискретна математика»[1] і практикума для студентів спеціальності 6.040301 «Прикладна математика»[2], в якому будуть присутні як тестові завдання так і практичні. Нижче, на рис. 1.1 – 1.3 приведені приклади які буде розв'язано в тренажері.

Застосувавши закони логіки висловлювань, доведемо еквівалентність формул $p \rightarrow (q \wedge r)$ і $(p \rightarrow q) \wedge (p \rightarrow r)$. Запишемо послідовність перетворень і назви використаних законів і правил:

$$\begin{aligned} p \rightarrow (q \wedge r) &= \bar{p} \vee (q \wedge r) \\ &\text{(за правилом усунення імплікації)} \\ &= (\bar{p} \vee q) \wedge (\bar{p} \vee r) \\ &\text{(за законом дистрибутивності 3a)} \\ &= (p \rightarrow q) \wedge (p \rightarrow r) \\ &\text{(за правилом введення імплікації).} \end{aligned}$$

За допомогою законів логіки висловлювань доведемо еквівалентність формул $p \rightarrow q$ та $\bar{q} \rightarrow \bar{p}$. Цю еквівалентність називають *правилом контрапозиції*.

$$\begin{aligned} p \rightarrow q &= \bar{p} \vee q \\ &\text{(за правилом усунення імплікації)} \\ &= q \vee \bar{p} \\ &\text{(за законом комутативності 1a)} \\ &= \overline{\bar{q}} \vee \bar{p} \\ &\text{(за законом подвійного заперечення б)} \\ &= \bar{q} \rightarrow \bar{p} \\ &\text{(за правилом введення імплікації).} \end{aligned}$$

Рисунок 1.1 - Приклади для розв'язування.

Розглянемо формули $f_1 = (p \rightarrow q)$, $f_2 = \bar{q}$, $g = \bar{p}$. Доведемо, що формула g – логічний наслідок формул f_1 і f_2 .

Спосіб 1. Скористаємося таблицями істинності, щоб показати, що формула g виконується в кожній інтерпретації, у якій виконується формула $(p \rightarrow q) \wedge \bar{q}$. Із табл. 1.10 видно, що є лише одна інтерпретація, у якій $(p \rightarrow q) \wedge \bar{q}$ виконується, а саме $p = F$, $q = F$; у цій інтерпретації формула \bar{p} також виконується. Отже, за означенням формула \bar{p} – логічний наслідок формул $p \rightarrow q$ та \bar{q} .

Таблиця 1.10

p	q	$p \rightarrow q$	\bar{q}	$(p \rightarrow q) \wedge \bar{q}$	\bar{p}
T	T	T	F	F	F
T	F	F	T	F	F
F	T	T	F	F	T
F	F	T	T	T	T

Рисунок 1.2 – Приклади для розв’язування

Спосіб 2. Скористаємося теоремою 1.2. Доведемо, що формула $(f_1 \wedge f_2) \rightarrow g$ загальнозначуща. Для цього побудуємо табл. 1.11 для формули $(f_1 \wedge f_2) \rightarrow g = ((p \rightarrow q) \wedge \bar{q}) \rightarrow \bar{p}$.

Таблиця 1.11

p	q	$((p \rightarrow q) \wedge \bar{q}) \rightarrow \bar{p}$
T	T	T
T	F	T
F	T	T
F	F	T

Оскільки формула $((p \rightarrow q) \wedge \bar{q}) \rightarrow \bar{p}$ загальнозначуща, то формула \bar{p} – логічний наслідок формул $p \rightarrow q$ та \bar{q} .

Спосіб 3. Скористаємося теоремою 1.3 та доведемо, що формула

$$(f_1 \wedge f_2) \rightarrow \bar{g} = ((p \rightarrow q) \wedge \bar{q}) \wedge \bar{\bar{p}} = ((p \rightarrow q) \wedge \bar{q}) \wedge p$$

заперечувана. Побудуємо таблицю істинності для цієї формули. Із табл. 1.12 видно, що формула $(p \rightarrow q) \wedge \bar{q} \wedge p$ хибна в будь-якій інтерпретації.

Таблиця 1.12

p	q	$p \rightarrow q$	\bar{q}	$(p \rightarrow q) \wedge \bar{q}$	$((p \rightarrow q) \wedge \bar{q}) \wedge p$
T	T	T	F	F	F
T	F	F	T	F	F
F	T	T	F	F	F
F	F	T	T	T	F

Тепер відповідно до теореми 1.3 можна дійти висновку, що формула \bar{p} логічно випливає з формул $p \rightarrow q$ та \bar{q} .

Рисунок 1.3 – Приклади для розв’язування

Наведені задачі (рис. 1.1) будуть розбиті на кроки. Студенту буде надана можливість самостійно розв’язати наведені задачі, шляхом перетягування відповідних символів логіки на відповідне місце в прикладові.

Задачі на рис. 1.2 – 1.3 будуть вже розв’язані, як приклад для пояснення студенту.

2 ІНФОРМАЦІЙНИЙ ОГЛЯД

2.1 Огляд робіт, де розглянуті тренажери для розв'язування оптимізаційних задач

Було проаналізовано з [5-8] роботи студентів-магістрів і бакалаврів ПУЕТ з розробки навчальних тренажерів для дистанційного курсу. Були переглянуті роботи таких студентів: Кондрашев Д. М. Розробка тренажера дистанційного навчального курсу «Математична логіка та теорія алгоритмів» з теми «Машини Т'юрінга»[9], Потерайла О. О. Програмування навчального тренажера для градієнтного методу оптимізації нелінійних функцій дистанційного навчального курсу «Методи оптимізації та дослідження операцій»[10], Русіна В. С. Програмна реалізація елементів тренажеру з теми "Аналіз алгоритму сортування вставками" дисципліни "Аналіз алгоритмів" [11] та інші[12 – 24].

2.2 Позитивні аспекти оглянутих робіт

Проаналізувавши роботи [9-24] по створенню навчальних тренажерів для дистанційного курсу, було виявлено такі позитивні сторони:

- проекти були виконані на таких мовах програмування, як Java, Ruby, C# та JavaScript, що дозволяє бути кроссплатформенними та дає змогу впроваджувати їх в дистанційний курс, за потреби;
- тренажери дають можливість в тій чи іншій мірі отримати знання і навички тем, які реалізовані в тренажерові;
- можна покращити навчання;
- за допомогою вивчень матеріалів в деяких тренажерах, можна збільшити отримання прибутку, зменшити затрати на виробництво та транспортування;
- покращують мислення студента;

- частково полегшують роботу викладача, а також вносять деяку новизну в вивчення предмету;
- великим плюсом є те, що дані тренажери існують.

2.3 Вади розробок з оглянутих робіт

Розглянуті програми також мають і недоліки, а саме:

- неможливість завантаження тренажера на ПК або мобільний пристрій;
- наявність підказок, навіть нескладні завдання, що не дає можливості студенту самому знаходити відповідь шляхом аналізу власних помилок;
- не всі тренажери маю функцію пропуску кроку, або повернення назад, що в свою чергу не дає можливості продовжити розглядання теми з того місця, де завершилася робота минулого разу;
- тренажери написані на одній мові, а саме українській, що зменшує можливість навчати іноземних студентів.

2.4 Необхідність та актуальність теми роботи

Розробка навчального тренажера з теми «Логіка висловлювань» для курсу «Математича логіка» допоможе студентам вивчити логічні оператори і операції з ними, допоможе розвитку мислення при програмуванні, а також допоможе закріпити отримані знання та навички отримані при вивченні курсу «Математична логіка».

Програма в випадках дистанційного або самостійного навчання заміняє викладача. Вона полегшить викладання практичного матеріалу, оскільки задачі, тести і теоретичний метеріал вже буде представлений в програмі, та його перевірку, що в деякій мірі зменшить навантаження на викладача, або дасть можливість обходитися без нього.

3. ТЕОРЕТИЧНА ЧАСТИНА

3.1. Алгоритмізація задачі за темою роботи

Спочатку перед студентом відкривається вікно з кнопкою меню, яка при натисканні розділяється на три кнопки вибору дій, і кнопкою підказки в верхньому правому кутку.

При виборі пункту меню що відповідає за перехід до навчальних матеріалів, відкривається екран, на якому присутні три кнопки: «Тестові завдання», «Практичні завдання», «Задачі з поясненням».

Крок 1.1. Натискаємо на кнопку «Тестові завдання». Перед студентом відкривається вікно, на якому ставиться питання: «Які з речень є висловлюваннями», і пропонується чотири варіанта відповіді А) на вулиці дощ. Б) викладач прийшов на урок. В) коли прийде Сергій? Г) Читай уважно! Після вибору відповідей, студент натискає на кнопку «Перевірити», якщо вибрані відповіді правильні, а саме А і Б то відбувається перехід до наступного кроку. Якщо відповідь не правильна, з'являється інформативне вікно про неправильність відповіді, студент робить відповідні висновки, вибирає правильну відповідь.

Крок 1.2. Крок 2-й. Перед студентом відкривається наступне вікно, де з'являється наступне завдання «Перетягніть знаки стосовно їхнім значенням». Студент повинен перетягнути відповідні знаки, а саме «логічне і», «логічне або», «логічне якщо ... то» до відповідних варіантів. Потім студент натискає кнопку відповідь і якщо все вірно переходить до наступного кроку, а якщо ні, програма сигналізує про не вірність спливаючим написом, студент робить висновки, робить правильне співвідношення знаків і переходить до наступного кроку.

Крок 1.3. Перед студентом відкривається наступне вікно, на якому ставиться питання «Вкажіть формулу закону суперечності» і пропонується чотири варіанта відповіді 1) $p \wedge \neg p = F$; 2) $p \vee \neg p = F$; 3) $p \vee q = q \vee p$; 4) $p \wedge p =$

p. Після вибору відповіді, студент натискає на кнопку «Відповідь», якщо вибрана відповідь правильна, а саме 1 то відбувається перехід до наступного кроку. Якщо відповідь не правильна, з'являється інформативний напис про неправильність відповіді, студент робить відповідні висновки, вибирає правильну відповідь.

Крок 1.4. Перед студентом відкривається наступне вікно, на якому ставиться питання « $p \rightarrow q = \neg p \vee q$ це» і пропонується чотири варіанта відповіді: 1) правило еквівалентності; 2) правило імплікації; 3) правило комутативності; 4) немає вірної відповіді. Після вибору відповіді, студент натискає на кнопку «Перевірити», якщо вибрана відповідь правильна, а саме 2 то відбувається перехід до наступного кроку. Якщо відповідь не правильна, з'являється інформативний напис про неправильність відповіді, студент робить відповідні висновки, вибирає правильну відповідь.

Крок 1.5. Перед студентом відкривається наступне вікно, на якому ставиться питання « $p \sim q = (p \rightarrow q) \wedge (q \rightarrow p)$ це» і пропонується чотири варіанта відповіді: 1) правило імплікації; 2) правило комутативності; 3) правило еквівалентності; 4) закон суперечності. Після вибору відповіді, студент натискає на кнопку «Перевірити», якщо вибрана відповідь правильна, а саме 3, то відбувається перехід до наступного кроку. Якщо відповідь не правильна, з'являється інформативний напис про неправильність відповіді, студент робить відповідні висновки, вибирає правильну відповідь.

Крок 1.6. Перед студентом відкривається наступне вікно, на якому ставиться питання « $p \vee (q \wedge r) = (p \vee q) \wedge (p \vee r)$ це» і пропонується чотири варіанта відповіді: 1) закон комутативності; 2) правило інтерполяції; 3) закон дистрибутивності; 4) закон поглинання. Після вибору відповіді, студент натискає на кнопку «Перевірити», якщо вибрана відповідь правильна, а саме 3, то відбувається перехід до наступного кроку. Якщо відповідь не правильна з'являється інформативний напис про неправильність відповіді, студент робить відповідні висновки і вибирає правильну відповідь.

Крок 1.7. Перед студентом відкривається наступне вікно, на якому ставиться питання: «Якщо вологість велика та температура висока, то ми не відчуваємо себе добре» представлено складним висловлюванням. Де воно?» і пропонується чотири варіанта відповіді: 1) $((p \wedge q) \rightarrow (\neg r))$; 2) $p \rightarrow r \wedge \neg q$; 3) $p \wedge q \rightarrow \neg r$; 4) $p \wedge r \rightarrow \neg q$. Після вибору відповіді, студент натискає на кнопку «Відповідь», якщо вибрана відповідь правильна, а саме 1, то відбувається перехід до наступного кроку. Якщо відповідь не правильна з'являється інформативний напис про неправильність відповіді, студент робить відповідні висновки і вибирає правильну відповідь.

Крок 1.8. Перед студентом відкривається наступне вікно, на якому ставиться питання: «Атом у логіці висловлювань це —» і пропонується чотири варіанта відповіді: 1) найменша частина виразу; 2) елемент рівняння; 3) формула; 4) правильна відповідь відсутня. Після вибору відповіді, студент натискає на кнопку «Перевірити», якщо вибрана відповідь правильна, а саме 3, то відбувається перехід до наступного кроку. Якщо відповідь не правильна з'являється інформативний напис про неправильність відповіді, студент робить відповідні висновки і вибирає правильну відповідь.

Крок 1.9. Перед студентом відкривається наступне вікно, на якому ставиться питання: «Результат застосування правила імплікації до формули $p \rightarrow q$ є» і пропонується чотири варіанта відповіді: 1) $\neg p \vee q$; 2) $\neg p \wedge q$; 3) $p \vee \neg q$; 4) $p \wedge q$. Після вибору відповіді, студент натискає на кнопку «Перевірити», якщо вибрана відповідь правильна, а саме 1, то відбувається перехід до наступного кроку. Якщо відповідь не правильна з'являється інформативний напис про неправильність відповіді, студент робить відповідні висновки і вибирає правильну відповідь.

Крок 1.10. Перед студентом відкривається наступне вікно, на якому ставиться питання: «Сформулюйте назву закону $\neg(\neg p) = p$ » і пропонується чотири варіанта відповіді: 1) закон подвійного заперечення; 2) закон усунення імплікації; 3) закон еквівалентності; 4) закон подвійної рівності. Після вибору відповіді, студент натискає на кнопку «Перевірити», якщо вибрана відповідь

правильна, а саме 1, то впливає вікно з інформацією про завершення тесту і кнопкою для переходу до головного меню. Якщо відповідь не правильна з'являється інформативний напис про неправильність відповіді, студент робить відповідні висновки і вибирає правильну відповідь.

Крок 2.1. В меню вибору завдань натискаємо кнопку «Практичні завдання». Після цього перед студентом висвітлюється вікно з вибором задачі «Задача №1», «Задача №2». При виборі задачі №1 переходимо до її розв'язання.

Крок 2.1.1. Перед студентом відкривається вікно на якому представлена задача: «Доведіть еквівалентність формули: $p \rightarrow (q \wedge r)$ і $(p \rightarrow q) \wedge (p \rightarrow r)$ Запишіть послідовність перетворень і виберіть назву використаного закону або правила». Нижче під умовою знаходиться вираз з пропусками які треба заповнити шляхом перетягування елементів (\wedge , \vee , \neg , \rightarrow) з лівого нижнього кутка в місце що виділено жовтим кольором $p \dots (q \wedge r) = \dots \dots (q \dots r)$. Після перетягування відповідних символів на відповідні місця, студент натискає кнопку «Показати варіанти відповіді». Після натискання з'являються три варіанти відповіді 1) За правилом усунення імплікації; 2) За законом дистрибутивності; 3) За правилом комутативності. Після вибору відповіді, студент натискає на кнопку «Перевірити», якщо вибрана відповідь правильна, а саме 1, то відбувається перехід до наступного кроку. Якщо відповідь не правильна з'являється інформативний напис про неправильність відповіді, студент робить відповідні висновки і вибирає правильну відповідь.

Крок 2.1.2 Перед студентом відкривається вікно на якому представлена задача: «Доведіть еквівалентність формули: $p \rightarrow (q \wedge r) = \neg p \vee (q \wedge r)$ Запишіть послідовність перетворень і виберіть назву використаного закону або правила». Нижче під умовою знаходиться вираз з пропусками які треба заповнити шляхом перетягування елементів (\wedge , q , \neg , r) з лівого нижнього кутка в місце що виділено жовтим кольором $(\dots \vee \dots) \wedge (\neg p \vee \dots)$. Після перетягування відповідних символів на відповідні місця, студент натискає кнопку «Показати варіанти відповіді». Після натискання з'являються три варіанти відповіді 1) За правилом усунення імплікації; 2) За законом дистрибутивності 3а; 3) За

правилом комутативності. Після вибору відповіді, студент натискає на кнопку «Перевірити», якщо вибрана відповідь правильна, а саме 2, то відбувається перехід на наступного кроку. Якщо відповідь не правильна з'являється інформативний напис про неправильність відповіді, студент робить відповідні висновки і вибирає правильну відповідь.

Крок 2.1.3. Перед студентом відкривається вікно на якому представлена задача: «Доведіть еквівалентність формули: $(\neg p \vee q) \wedge (\neg p \vee r)$ Запишіть послідовність перетворень і виберіть назву використаного закону або правила». Нижче під умовою знаходиться вираз з пропусками які треба заповнити шляхом перетягування елементів (\wedge , \vee , \rightarrow , \leftrightarrow) з лівого нижнього кутка в місце що виділено жовтим кольором $(p \dots q) \dots (p \dots r)$. Після перетягування відповідних символів на відповідні місця, студент натискає кнопку «Показати варіанти відповіді». Після натискання з'являються три варіанти відповіді 1) За правилом усунення імплікації; 2) За правилом дистрибутивності; 3) За правилом комутативності. Після вибору відповіді, студент натискає на кнопку «Перевірити», якщо вибрана відповідь правильна, а саме 1, то висвітлюється вікно, що інформує про завершення розв'язку задачі і надається два варіанти вибору, або повернутися до головного меню, або перейти до вирішення задачі №2. Якщо відповідь не правильна з'являється інформативний напис про неправильність відповіді, студент робить відповідні висновки і вибирає правильну відповідь.

Крок 2.2. В меню вибору завдань натискаємо кнопку «Практичні завдання». Після цього перед студентом висвітлюється вікно з вибором задачі «Задача №1», «Задача №2». При виборі задачі №2 переходимо до її розв'язання.

Крок 2.2.1. Перед студентом відкривається вікно на якому представлена задача: «Доведіть еквівалентність формули: $p \rightarrow q$ та $\neg q \rightarrow \neg p$. Запишіть послідовність перетворень і виберіть назву використаного закону або правила». Нижче під умовою знаходиться вираз з пропусками які треба заповнити шляхом перетягування елементів (\wedge , \vee , $=$, \rightarrow) з лівого нижнього кутка в місце що виділено жовтим кольором $p \dots q \dots \neg p \dots q$. Після перетягування відповідних

символів на відповідні місця, студент натискає кнопку «Показати варіанти відповіді». Після натискання з'являються три варіанти відповіді 1) За правилом усунення імплікації; 2) За законом дистрибутивності; 3) За правилом комутативності. Після вибору відповіді, студент натискає на кнопку «Перевірити», якщо вибрана відповідь правильна, а саме 1, то відбувається перехід до наступного кроку. Якщо відповідь не правильна з'являється інформативний напис про неправильність відповіді, студент робить відповідні висновки і вибирає правильну відповідь.

Крок 2.2.2. Перед студентом відкривається вікно на якому представлена задача: «Доведіть еквівалентність формули: $p \rightarrow q = \neg p \vee q$. Запишіть послідовність перетворень і виберіть назву використаного закону або правила». Нижче під умовою знаходиться вираз з пропусками які треба заповнити шляхом перетягування елементів (\wedge , \vee , $=$, \rightarrow) з лівого нижнього кутка в місце що виділено жовтим кольором $q \dots \neg p$. Після перетягування відповідних символів на відповідні місця, студент натискає кнопку «Показати варіанти відповіді». Після натискання з'являються три варіанти відповіді 1) За законом дистрибутивності; 2) За законом дистрибутивності 3а; 3) За законом комутативності 1а. Після вибору відповіді, студент натискає на кнопку «Перевірити», якщо вибрана відповідь правильна, а саме 3, то відбувається перехід до наступного кроку. Якщо відповідь не правильна з'являється інформативний напис про неправильність відповіді, студент робить відповідні висновки і вибирає правильну відповідь.

Крок 2.2.3. Перед студентом відкривається вікно на якому представлена задача: «Доведіть еквівалентність формули: $q \vee \neg p$. Запишіть послідовність перетворень і виберіть назву використаного закону або правила». Нижче під умовою знаходиться вираз з пропусками які треба заповнити шляхом перетягування елементів (q , p , $\neg p$, $\neg(\neg q)$) з лівого нижнього кутка в місце що виділено жовтим кольором $\dots \vee \dots$. Після перетягування відповідних символів на відповідні місця, студент натискає кнопку «Показати варіанти відповіді». Після натискання з'являються три варіанти відповіді 1) За законом дистрибутивності

За; 2) За законом подвійного заперечення; 3) За правилом комутативності. Після вибору відповіді, студент натискає на кнопку «Перевірити», якщо вибрана відповідь правильна, а саме 2, то відбувається перехід до наступного кроку. Якщо відповідь не правильна з'являється інформативний напис про неправильність відповіді, студент робить відповідні висновки і вибирає правильну відповідь.

Крок 2.2.4. Перед студентом відкривається вікно на якому представлена задача: «Доведіть еквівалентність формули: $\neg(\neg q) \vee \neg p$. Запишіть послідовність перетворень і виберіть назву використаного закону або правила». Нижче під умовою знаходиться вираз з пропусками які треба заповнити шляхом перетягування елементів ($\neg q$, p , $\neg p$, $\neg(\neg q)$) з лівого нижнього кутка в місце що виділено жовтим кольором $\dots \rightarrow \dots$. Після перетягування відповідних символів на відповідні місця, студент натискає кнопку «Показати варіанти відповіді». Після натискання з'являються три варіанти відповіді 1) За правилом введення імплікації; 2) За правилом усунення імплікації; 3) За законом дистрибутивності. Після вибору відповіді, студент натискає на кнопку «Перевірити», якщо вибрана відповідь правильна, а саме 1, то висвітлюється вікно, що інформує про завершення розв'язку задачі і надається два варіанти вибору, або повернутися до головного меню, або перейти до вирішення ще одної задачі (задача №1). Якщо відповідь не правильна з'являється інформативний напис про неправильність відповіді, студент робить відповідні висновки і вибирає правильну відповідь.

Крок 3.1. В меню вибору завдань натискаємо кнопку «Задача з поясненням». Після цього перед студентом висвітлюється вікно з вибором: «Задача №3», «Теореми». При виборі задачі №3 переходимо до її розв'язання. Задачу №3 взято з [1].

Крок 3.1.1. Перед студентом відкривається екран з заголовком «Розглянемо формули $F1 = (P \rightarrow Q)$, $F2 = \neg Q$, $G = \neg P$ ». Нижче написано: «Доведемо, що формула G – логічний наслідок формули $F1$ і $F2$. Нижче присутня кнопка «Розв'язання». При її натисканні висвітлюється текст

пояснення: «Спосіб 1. Скористаємося таблицями істинності. Щоб показати, що формула G виконується в кожній інтерпритації, у якій виконується формула $(P \rightarrow Q) \wedge \neg Q$ », і таблиця істинності, яка забарвлена в 4 кольори, зелений, червоний, сірий, білий. Сірий це шапка таблиці, білий це варіанти «Т» або «F» для елементів «Р» та «Q», що знаходяться в двох лівих колонках. Червоний та зелений це індикатор правильності відповідно семантиці (рис. 3.1). Зелений це вірно, червоний — ні. При натисканні на кожен з червоних або зелених клітинок, висвітлюється пояснення чому саме так (вірно або не вірно) згідно семантиці.

P	Q	$P \rightarrow Q$		$\neg Q$		$(P \rightarrow Q) \wedge \neg Q$		$\neg P$	
T	T	T	F	T	F	T	F	T	F
T	F	T	F	T	F	T	F	T	F
F	T	T	F	T	F	T	F	T	F
F	F	T	F	T	F	T	F	T	F

Рисунок 3.1. Приклад таблиці в тренажері.

Під таблицею знаходиться пояснення: «Із таблиці видно, що є лише одна інтерпритація, у якій $(P \rightarrow Q) \wedge \neg Q$ виконується а саме в $P = F, Q = F$; у цій інтерпритації формула $\neg P$ також виконується. Отже, за означеннями формула $\neg P$ – логічний наслідок формул $P \rightarrow Q$ та $\neg Q$ ». Нижче під поясненням знаходиться кнопка далі, що переносить до способу 2.

Крок 3.1.2. Перед студентом відкривається екран з заголовком «Спосіб 2». Нижче написано «Скористаймося теоремою 1 (що присутня при виборі в головному меню на крокові 3.1). Доведемо, що формула $F1 \wedge F2 \rightarrow G$ загальнозначуща. Для цього побудуємо таблицю для формули формула $F1 \wedge F2 \rightarrow G = ((P \rightarrow Q) \wedge \neg Q) \rightarrow \neg P$ ». Нижче присутня кнопка «Розв’язання». При її

натисканні висвітлюється таблиця істинності (рис. 3.2) і текст пояснення «Оскільки формула $((P \rightarrow Q) \wedge \neg Q) \rightarrow \neg P$ загальнозначуща, то формула $\neg P$ - логічний наслідок формул $P \rightarrow Q$ та $\neg Q$.».

P	Q	$((P \rightarrow Q) \wedge \neg Q) \rightarrow \neg P$	
T	T	T	F
T	F	T	F
F	T	T	F
F	F	T	F

Рисунок 3.2. Приклад таблиці для способу 2.

Нижче під поясненням знаходиться кнопка далі, що переносить до способу 3.

Крок 3.1.3. Перед студентом відкривається екран з заголовком «Спосіб 3». Нижче написано «Скористаємося теоремою та доведемо що формула $(F1 \wedge F2) \rightarrow \neg G = ((P \rightarrow Q) \wedge \neg Q) \wedge \neg(\neg P) = ((P \rightarrow Q) \wedge \neg Q) \wedge P$ заперечувана. Побудуємо таблицю істинності для цієї формули. Із таблиці видно, що формула $((P \rightarrow Q) \wedge \neg Q) \rightarrow \neg P$ хибна в будь-якій інтерпретації.». Нижче присутня кнопка «Розв’язання». При її натисканні висвітлюється таблиця істинності (рис. 3.3) і текст пояснення: «Тепер відповідно до теорми можна дійти висновку, що формула $\neg P$ логічно випливає з формул $P \rightarrow Q$ та $\neg Q$.». Нижче знаходиться кнопка «До головного меню» при натисканні на яку користувач переходить до вибору навчальних матеріалів які описані на початку алгоритма.

P	Q	$P \rightarrow Q$		$\neg Q$		$(P \rightarrow Q) \wedge \neg Q$		$((P \rightarrow Q) \wedge \neg Q) \wedge P$	
T	T	T	F	T	F	T	F	T	F
T	F	T	F	T	F	T	F	T	F
F	T	T	F	T	F	T	F	T	F
F	F	T	F	T	F	T	F	T	F

Рисунок 3.3 Приклад таблиці істинності для способу 3.

Крок 3.2.1 В меню вибору завдань, натискаємо на кнопку «Задачі з поясненням», вибираємо розділ «Теореми». Після натискання переходимо до самих теорем.

Крок 3.2.2. Перед студентом з'являється екран з заголовком «Теорема 1» і сама теорема нижче: «Формула G - логічний наслідок формул F_1, F_2, \dots, F_n тоді й лише тоді, коли формула $((F_1 \wedge F_2 \wedge \dots \wedge F_n) \rightarrow G)$ - загально значуща. Доведення теорема. Необхідність. Нехай G - логічний наслідок формул F_1, F_2, \dots, F_n та I - довільна їх інтерпретація. Якщо формули F_1, F_2, \dots, F_n істинні в інтерпретації I , то за означенням логічного наслідку формула G також істинна в I . Звідси випливає, що формула $((F_1 \wedge F_2 \wedge \dots \wedge F_n) \rightarrow G)$ істинна в інтерпретації I . З іншого боку, якщо не всі формули з F_1, F_2, \dots, F_n істинні в інтерпретації I , тобто принаймні одна з них хибна в I , то формула $((F_1 \wedge F_2 \wedge \dots \wedge F_n) \rightarrow G)$ істинна в I . Отже, формула $((F_1 \wedge F_2 \wedge \dots \wedge F_n) \rightarrow G)$ істинна в довільній інтерпретації, або $\models ((F_1 \wedge F_2 \wedge \dots \wedge F_n) \rightarrow G)$. Достатність. Припустимо що формула $((F_1 \wedge F_2 \wedge \dots \wedge F_n) \rightarrow G)$ загальнозначуща. Тоді Якщо формула $(F_1 \wedge F_2 \wedge \dots \wedge F_n)$ істинна в якійсь інтерпретації, то й формула G має бути істинною в цій інтерпретації, тобто G - логічний наслідок формул F_1, F_2, \dots, F_n . Якщо формула G - логічний наслідок формул F_1, F_2, \dots, F_n то формулу $((F_1 \wedge F_2 \wedge \dots \wedge F_n) \rightarrow G)$ називають логічною теоремою, а G - вивідна формула. Вираз $F_1,$

$F_2, \dots, F_n \vdash G$ називають правилом виведення. Тут гіпотези записано зліва від знака \vdash , а висновок - справа. Сам знак \vdash має зміст "отже".»

Нижче видно підказку «Листати тут \rightarrow », що повідомляє про те, що для того щоб перейти на наступну сторінку треба провести пальцем по екрану з права на ліво і перейти до наступної теореми.

Крок 3.2.3. Перед студентом з'являється екран з заголовком «Теорема 2 (принцип прямої дедукції)» і сама теорема нижче: «Формула G - логічний наслідок формул F_1, F_2, \dots, F_n тоді й лише тоді, коли формула $((F_1 \wedge F_2 \wedge \dots \wedge F_n) \rightarrow G)$ - суперечність. \n\n За теоремою 1 формула G - логічний наслідок формул F_1, F_2, \dots, F_n тоді й лише тоді, коли формула $((F_1 \wedge F_2 \wedge \dots \wedge F_n) \rightarrow G)$ - загально значуща. Отже, G - логічний наслідок формул F_1, F_2, \dots, F_n тоді й лише тоді, коли заперечення формули $((F_1 \wedge F_2 \wedge \dots \wedge F_n) \rightarrow G)$ - суперечність. Справді,» після самої теореми нижче знаходиться картинка формули (рис. 3.4.).

$$\overline{(f_1 \wedge f_2 \wedge \dots \wedge f_n) \rightarrow g} = \overline{(f_1 \wedge f_2 \wedge \dots \wedge f_n) \vee g} = \\ = \overline{f_1 \vee f_2 \vee \dots \vee f_n \vee g} = f_1 \wedge f_2 \wedge \dots \wedge f_n \wedge \bar{g}.$$

Рисунок 3.4. Формула для теореми 2.

Нижче видно підказку « \leftarrow Листати тут», що повідомляє про те, що для того щоб перейти на наступну сторінку треба провести пальцем по екрану з права на ліво і перейти до попередньої теореми. Щоб повернутися до головного меню вибору завдань, треба натиснути кнопку «Назад» на вашому смартфоні.

3.2. Блок-схема

Нижче наведено блок-схему програми.

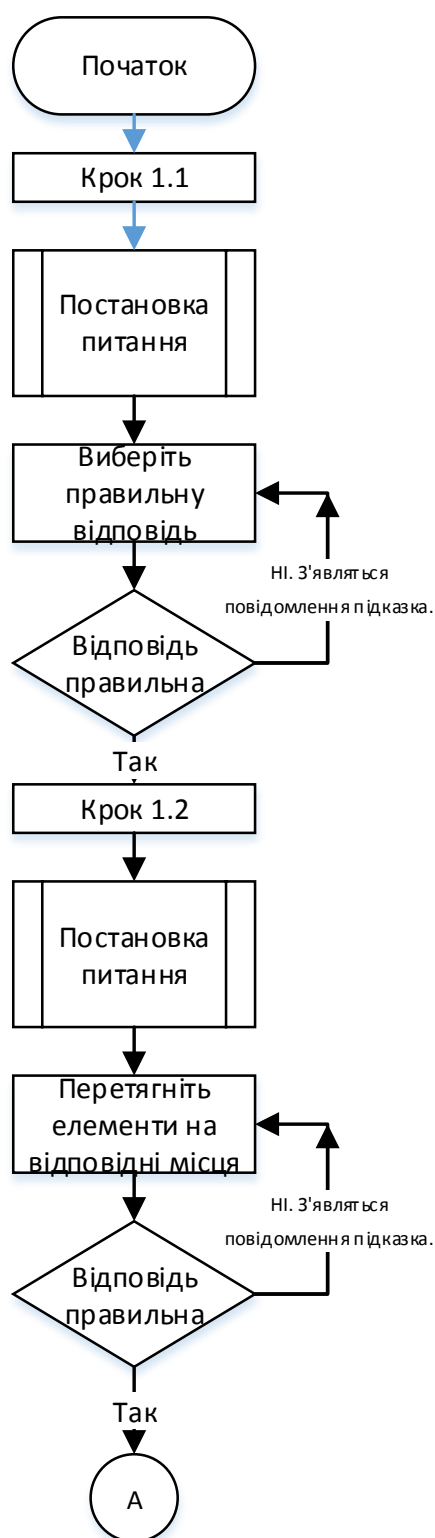


Рисунок 3.5 - Блок-схема тренажера (частина 1)

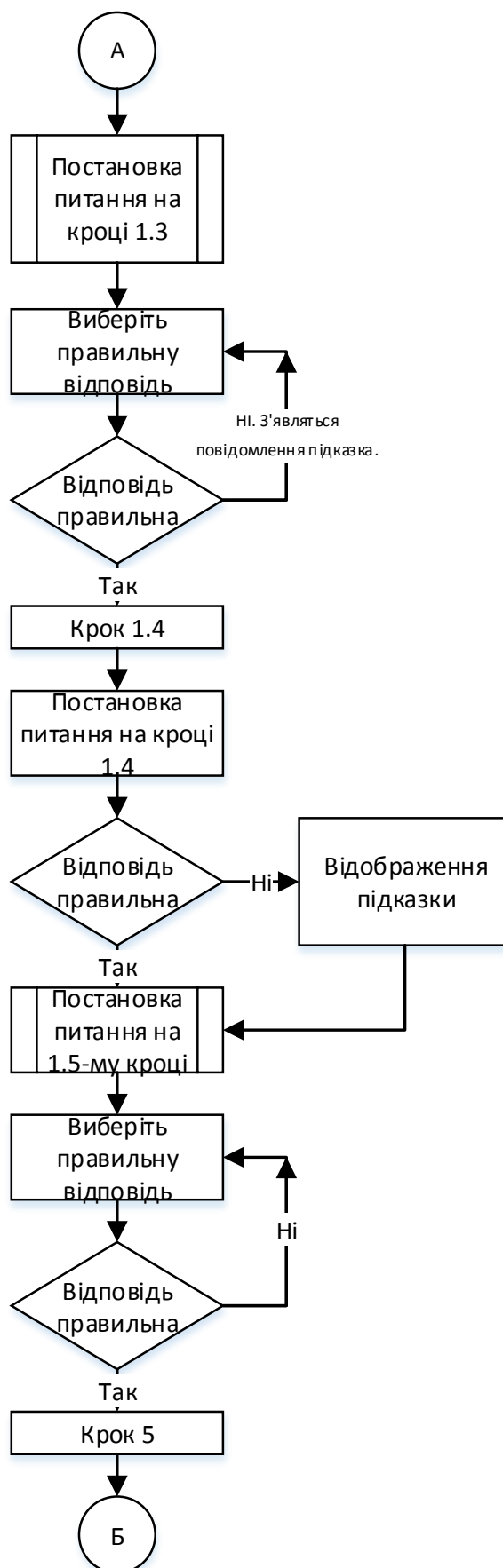


Рисунок 3.6 - Блок-схема тренажера (частина 2)

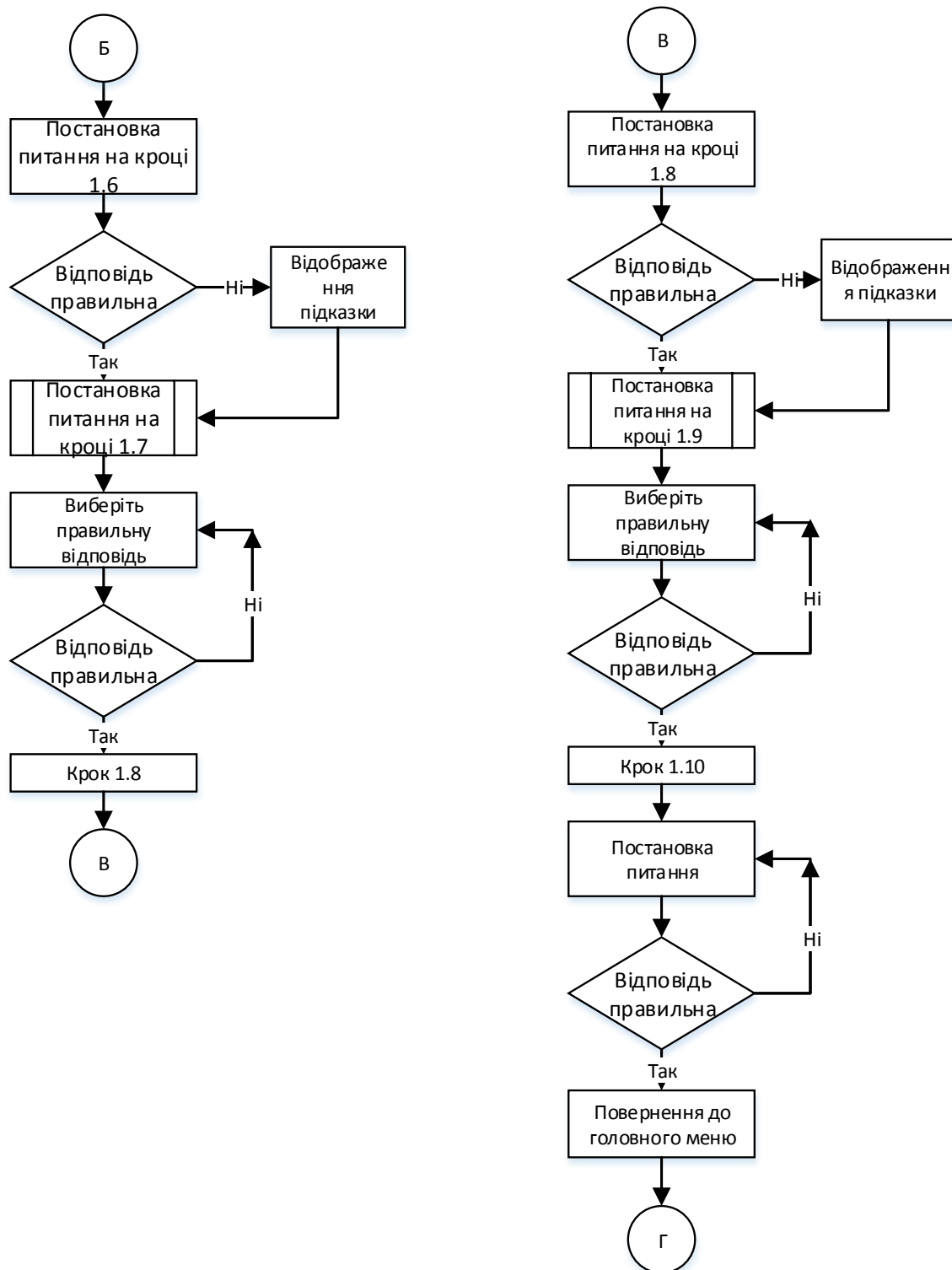


Рисунок 3.7 - Блок-схема тренажера (частина 3)

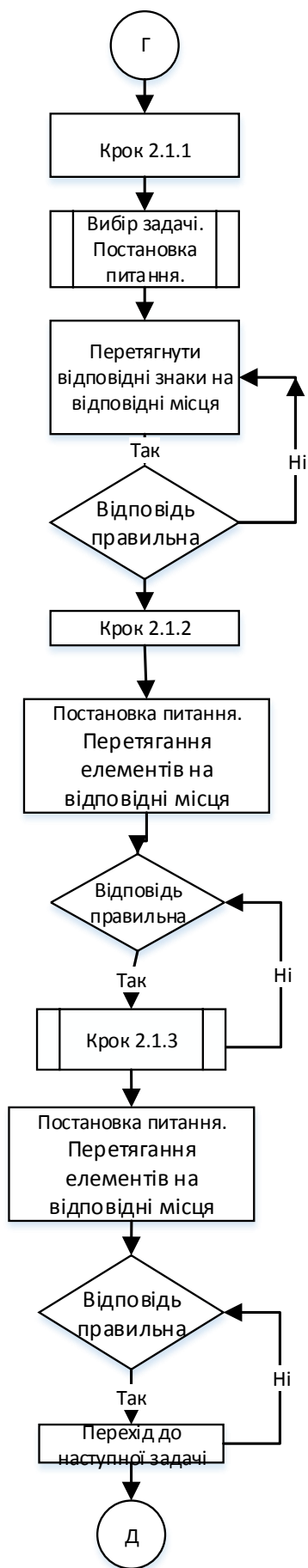


Рисунок 3.8 - Блок-схема тренажера (частина 4)



Рисунок 3.9 - Блок-схема тренажера (частина 5)

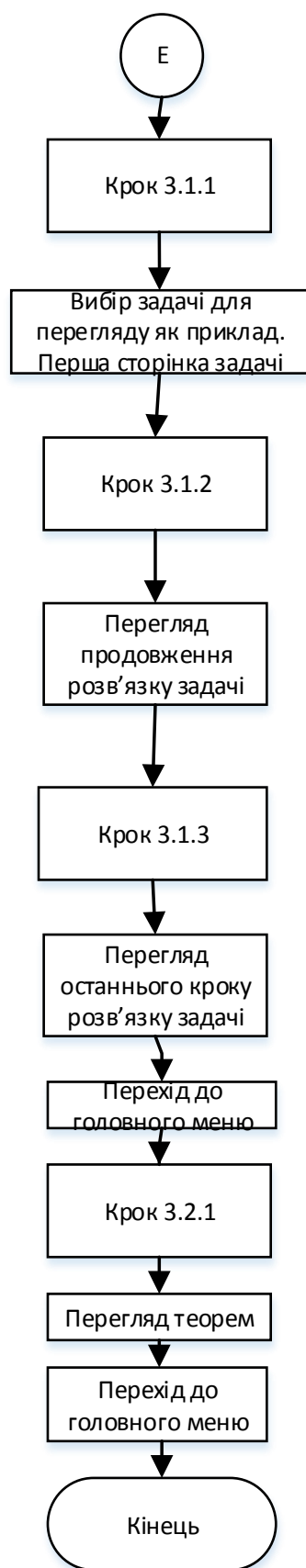


Рисунок 3.10 - Блок-схема тренажера (частина 6)

3.3. Обґрунтування вибору програмних засобів для реалізації завдання роботи

Для виконання поставленої задачі було застосовано високорівневу мову програмування Java. Середовищем розробки виступила програма Android Studio. Дана мова була обрана тому, що програму, яку на ній створюватиметься, можна встановити на Android-пристрої з версією android 7.0 і вище.

4. ПРАКТИЧНА ЧАСТИНА

4.1. Опис процесу програмної реалізації

Реалізація програми розпочато з вивчення матеріалу логіки висловлювань, види логічних операторів і тд [1].

Роботу розпочато з створення нового проекту Empty Activity і заповнення його елементами меню (рис. 4.1).

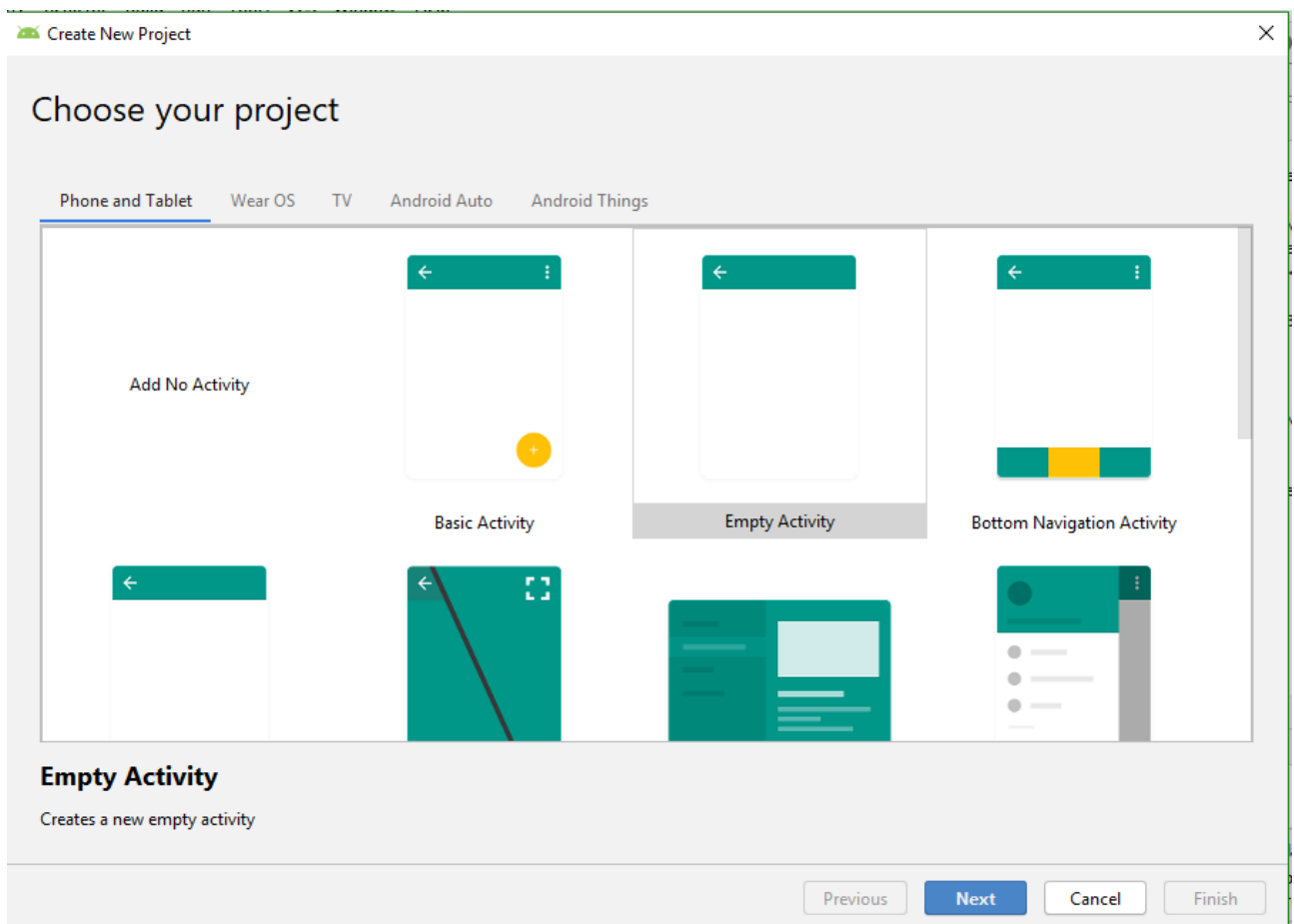


Рисунок 4.1 - Створення нової форми

Після створення проекту, заповнюємо її елементами, а саме, такими елементами як TextView, CheckBox, Radio Button, Button. Наявність тих чи інших елементів залежала від «кроку» програми.

Так як в тренажері проводиться перевірка завдань на вірність, перевірку було вирішено виконати через оператор if...else рис. 4.2.

Також було реалізовано спливаюче вікно, для індикації правильності відповіді за допомогою Toast. Приклад наведено на рис. 4.3.

```
else if (view.getId() == R.id.if1 && v.getId() == R.id.targetBox3) {
    LinearLayout oldPar = (LinearLayout) view.getParent();
    oldPar.removeView(view);
    LinearLayout newPar = (LinearLayout) v;
    box3.setVisibility(View.GONE);
    newPar.addView(view);
}
else {
    Toast toast = Toast.makeText(getApplicationContext(), R.string.prompt_2, Toast.LENGTH_SHORT);
    toast.show();
}
break;
```

Рисунок 4.2 - Приклад перевірки оператором if

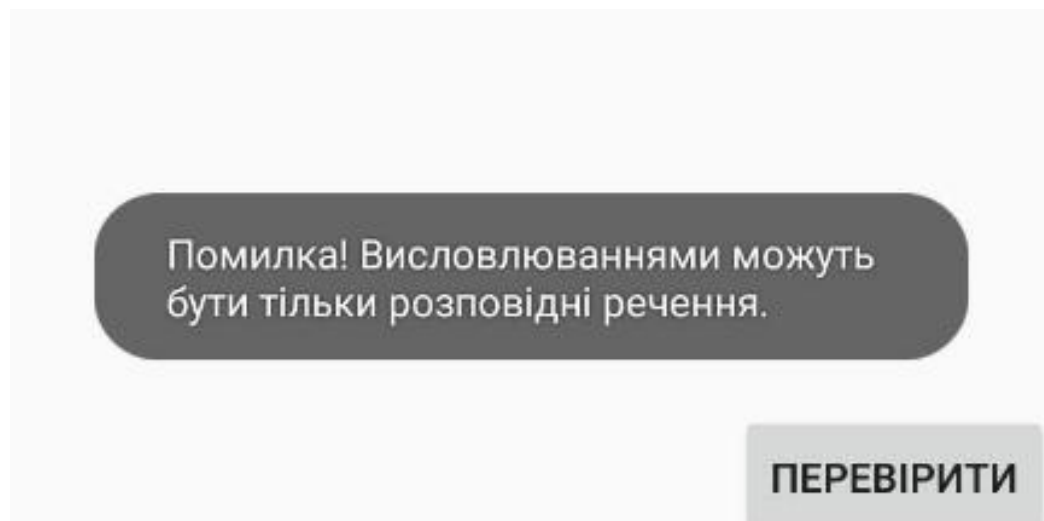


Рисунок 4.3 - Застосування Toast

Також було реалізована функція Drag and Drop, для перетягування елементів на відповідні місця рис. 4.4.


```

View.OnTouchListener touchClickListener = (v, event) -> {

    ClipData data = ClipData.newPlainText( label: "", text: "");
    View.DragShadowBuilder myShadowBuilder = new View.DragShadowBuilder(v);
    v.startDrag(data, myShadowBuilder,v, flags: 0);
    return true;
};
View.OnDragListener dragListener = (v, event) -> {

    int dragEvent = event.getAction();
    final View view = (View) event.getLocalState();

    switch (dragEvent){
        case DragEvent.ACTION_DRAG_ENTERED:
            break;
        case DragEvent.ACTION_DRAG_EXITED:
            break;
        case DragEvent.ACTION_DROP:

            if (view.getId()==R.id.and && v.getId()==R.id.targetBox1){
                LinearLayout oldPar = (LinearLayout) view.getParent();
                oldPar.removeView(view);
                LinearLayout newPar = (LinearLayout)v;
                box1.setVisibility(View.GONE);
                newPar.addView(view);
            }
    }
}

```

Рисунок 4.4. Приклад застосування Drag and Drop

4.2. Опис програми

На початку було створено меню тренажера яке розгорталось б при натисканні на нього на три кнопки: 1) кнопка для переходу до навчання; 2) кнопка для зміни мови (українська/англійська); 3) кнопка для переходу до інформації про розробника і керівника.(рис.4.5 – 4.6).

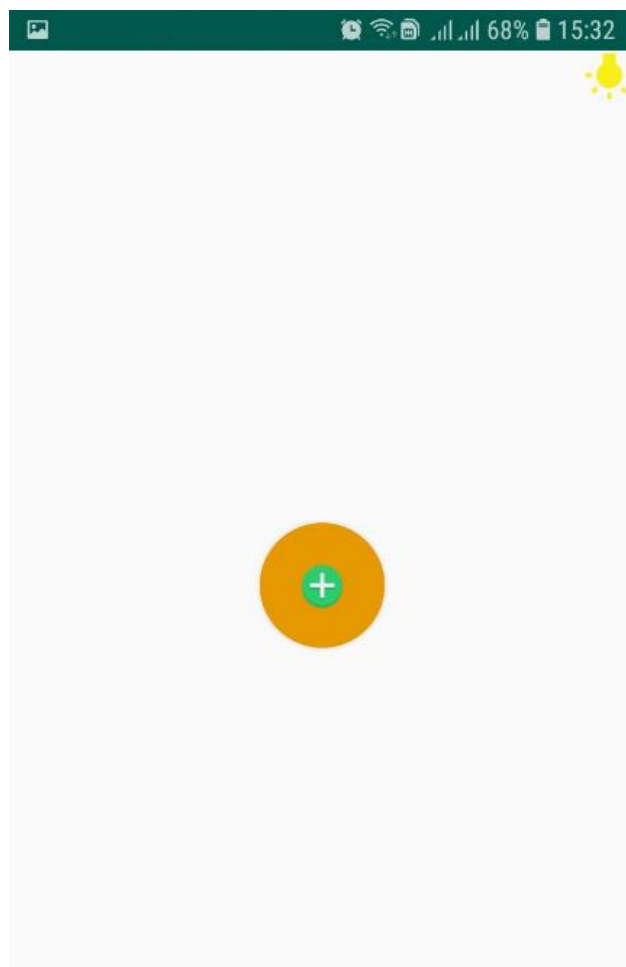


Рисунок 4.5 - Меню тренажера до натискання

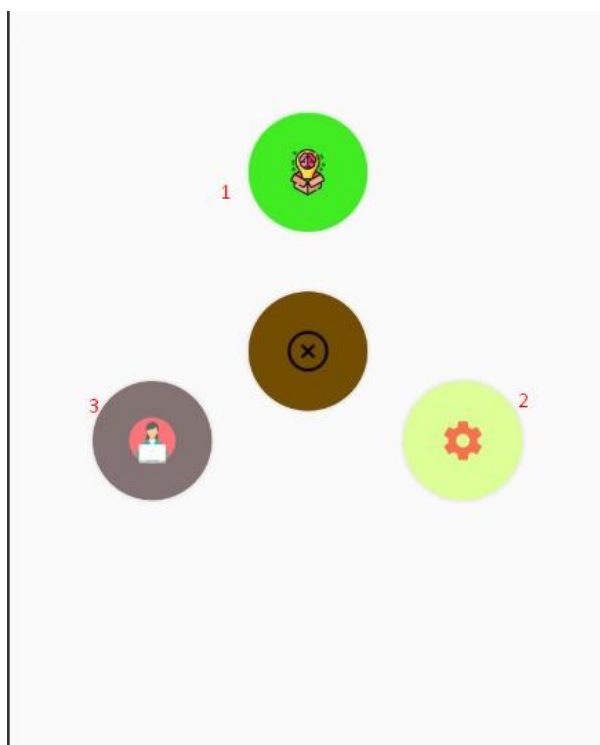


Рисунок 4.6 – Меню тренажера після розкриття

Після переходу до навчальної частини тренажера, на екрані смартфона відображаються три кнопки з вибором конкретного завдання, чи то тести, чи то практичне розв'язання задач, чи то розглядання задачі з поясненням рис. 4.7.



Рисунок 4.7. Екран з вибором типу завдань.

4.3. Перевірка валідності

Після розробки тренажера, він був протестований на кожному кроці на наявність помилок і відсутності багів.

На перших кроках (1.1 – 1.10) було протестовано всі варіанти відповідей. Для прикладу наведі скріншоти типових кроків тестових завдань (рис. 4.8 – 4.14).

Які з речень є висловлюванням?

1. На вулиці дощ.
2. Викладач не прийшов на урок.
3. Коли прийде Сергій?
4. Читай уважно!

☐ 1 ☐ 2 ☒ 3 ☐ 4

Помилка! Висловлюваннями можуть бути тільки розповідні речення.

ПЕРЕВІРИТИ

Рисунок 4.8 - Перевірка третього варіанту відповіді на першому кроці

Які з речень є висловлюванням?

1. На вулиці дощ.
2. Викладач не прийшов на урок.
3. Коли прийде Сергій?
4. Читай уважно!

☐ 1 ☐ 2 ☒ 3 ☒ 4

Помилка! Висловлюваннями можуть бути тільки розповідні речення.

ПЕРЕВІРИТИ

Рисунок 4.9 - Перевірка третього і четвертого варіанту відповіді на першому кроці

Які з речень є висловлюванням?

1. На вулиці дощ.
2. Викладач не прийшов на урок.
3. Коли прийде Сергій?
4. Читай уважно!

☒ 1 ☒ 2 ☐ 3 ☐ 4

ПЕРЕВІРИТИ

Рисунок 4.10 - Перевірка першого і другого варіантів відповіді на першому кроці

Перетягніть знаки стосовно їхнім значенням

Логічне і -

Логічне або -

Якщо...то -

Помилка! Зробіть правильну відповідність.

ПЕРЕВІРИТИ

Рисунок 4.11 Перетягування елементів на невідповідне місце.

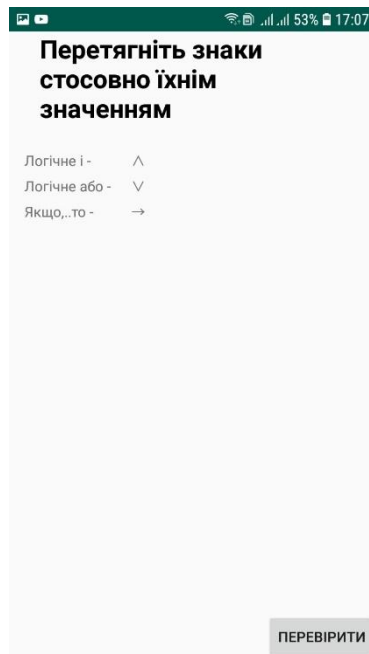


Рисунок 4.12 Перетягування елементів на відповідне місце.

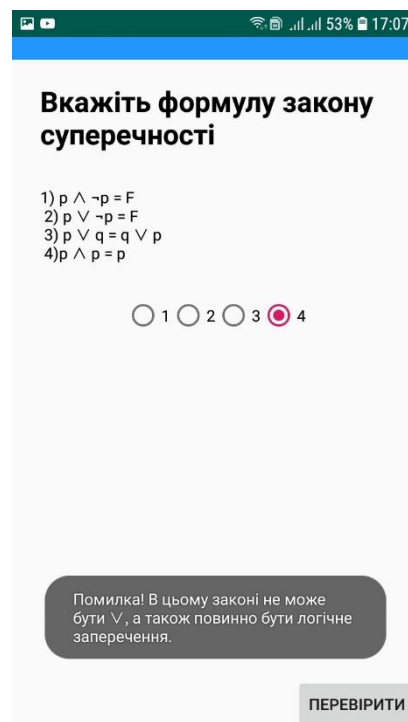
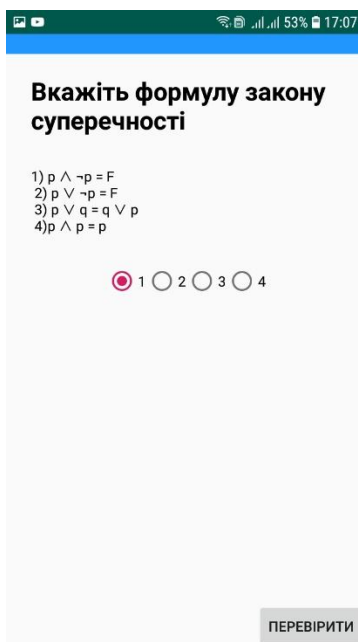


Рисунок 4.13. Вибір не правильної відповіді на кроці 3



Вкажіть формулу закону суперечності

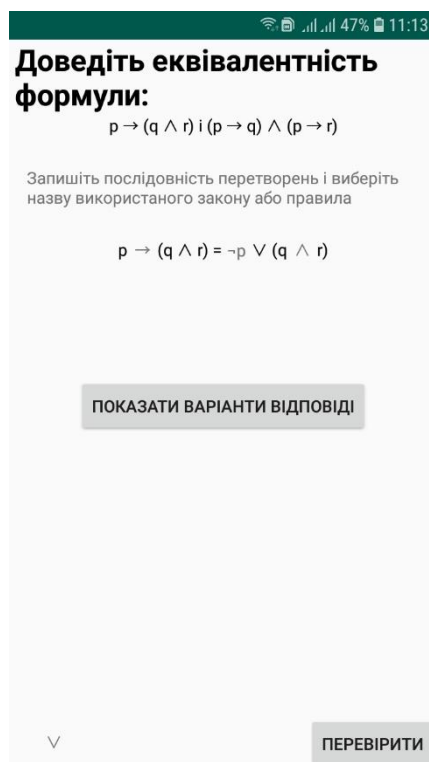
1) $p \wedge \neg p = F$
 2) $p \vee \neg p = F$
 3) $p \vee q = q \vee p$
 4) $p \wedge p = p$

☒ 1 ☐ 2 ☐ 3 ☐ 4

ПЕРЕВІРИТИ

Рисунок 4.14. Вибір правильної відповіді на кроці 3

Далі було протестовано кроки 2.1.1 – 2.2.4 на відповідність роботи і відсутність багів. На рис 4.15 – 4.16 продемонстровані приклади роботи типового кроку.



Доведіть еквівалентність формули:

$p \rightarrow (q \wedge r) \vee (p \rightarrow q) \wedge (p \rightarrow r)$

Запишіть послідовність перетворень і виберіть назву використаного закону або правила

$p \rightarrow (q \wedge r) = \neg p \vee (q \wedge r)$

ПОКАЗАТИ ВАРІАНТИ ВІДПОВІДІ

ПЕРЕВІРИТИ

Рисунок 4.15. Крок 2.1.1 з правильно перетягнутими елементами в формулі

Доведіть еквівалентність формули:

$$p \rightarrow (q \wedge r) \wedge (p \rightarrow q) \wedge (p \rightarrow r)$$

Запишіть послідовність перетворень і виберіть назву використаного закону або правила

$$p \rightarrow (q \wedge r) = \neg p \vee (q \wedge r)$$

☐ За правилом усунення імплікації
☒ За законом дистрибутивності
☐ За правилом комутативності

Помилка! Зробіть правильний вибір.

▼ ПЕРЕВІРИТИ

Рисунок 4.16. Крок 2.1.1 з неправильно вибраним варіантом відповіді в тестовому завданні після розв'язку прикладу

Також було протестовано кроки 3.1.1 – 3.2.3 на відсутність багів. Нижче продемонстрована одна з задач (рис. 4.17-4.20), а також теореми (рис 4.21 – 4.22), які використовуються в тренажері.

Розглянемо формули F1 = (P → Q), F2 = ¬Q, G = ¬P

Доведемо, що формула G - логічний наслідок формул F1 і F2

РОЗВ'ЯЗАННЯ

Рисунок 4.17. Задача на крокові 3.1.1. до перегляду розв'язання

Розглянемо формули $F1 = (P \rightarrow Q)$,
 $F2 = \neg Q$, $G = \neg P$

Доведемо, що формула G - логічний наслідок
 формул $F1$ і $F2$

Спосіб 1. Скористаємося таблицями істинності,
 щоб показати, що формула G виконується в кожній
 інтерпретації, у якій виконується формула
 $(P \rightarrow Q) \wedge \neg Q$

P	Q	$P \rightarrow Q$	$\neg Q$	$(P \rightarrow Q) \wedge \neg Q$	$\neg P$
T	T	T	F	F	F
T	F	F	T	F	F
F	T	T	F	F	T
F	F	T	T	T	T

Із таблиці видно, що є лише одна інтерпретація, у
 якій $(P \rightarrow Q) \wedge \neg Q$ виконується а саме $P = F$, $Q = F$;
 у цій інтерпретації формула $\neg P$ також виконується.
 Отже, за означенням формула $\neg P$ - логічний
 наслідок формул $P \rightarrow Q$ та $\neg Q$

ДАЛІ

Рисунок 4.18. Задача на крокові 3.1.1 після натискання кнопки
 розв'язання

Розглянемо формули $F1 = (P \rightarrow Q)$,
 $F2 = \neg Q$, $G = \neg P$

Доведемо, що формула G - логічний наслідок
 формул $F1$ і $F2$

Спосіб 1. Скористаємося таблицями істинності,
 щоб показати, що формула G виконується в кожній
 інтерпретації, у якій виконується формула
 $(P \rightarrow Q) \wedge \neg Q$

P	Q	$P \rightarrow Q$	$\neg Q$	$(P \rightarrow Q) \wedge \neg Q$	$\neg P$
T	T	T	F	F	F
T	F	F	T	F	F
F	T	T	F	F	T
F	F	T	T	T	T

Із таблиці видно, що є лише одна інтерпретація, у
 якій $(P \rightarrow Q) \wedge \neg Q$ виконується а саме $P = F$, $Q = F$;
 у цій інтерпретації формула $\neg P$ також виконується.
 Отже, за означенням формула $\neg P$ - логічний
 наслідок формул $P \rightarrow Q$ та $\neg Q$

ДАЛІ

Рисунок 4.19. Задача на крокові 3.1.1 після натискання коміркі зі знаком
 «Т» або «F»

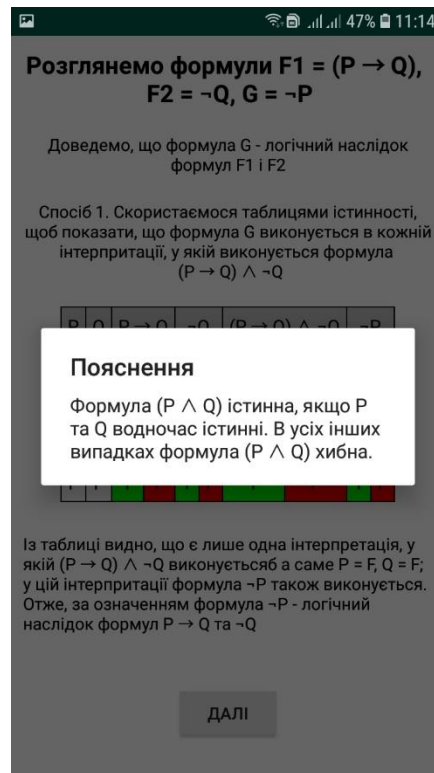


Рисунок 4.20. Висвітлення вікна з поясненням в задачі на крокові 3.1.1 після натискання на відповідну комірку зі знаком «Т» або «F»

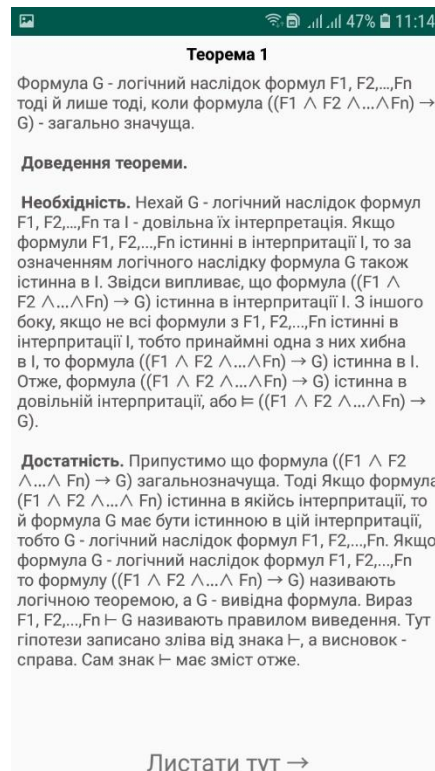


Рисунок 4.21. Теорема 1 на крокові 3.2.1

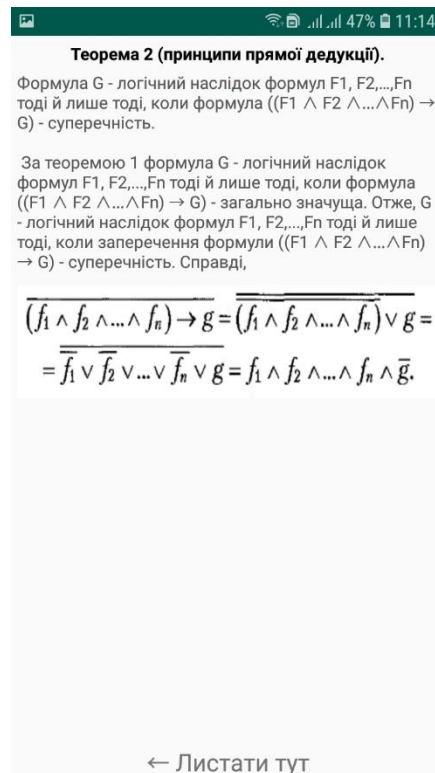


Рисунок 4.22. Теорема 2 на крокові 3.2.2

Коли користувач на останньому тесті або на останньому кроці розв'язку задачі вибрав правильну відповідь, йому пропонується повернутися до головного меню (рис. 4.23 - 4.24).

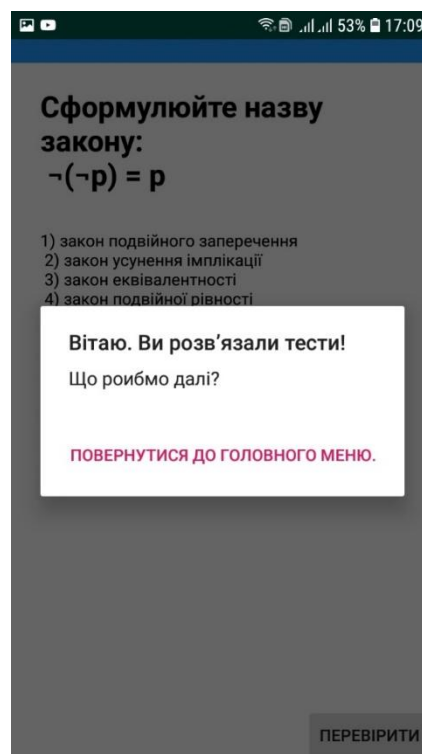


Рисунок 4.23. Перехід до головного меню після відповіді на останній тест

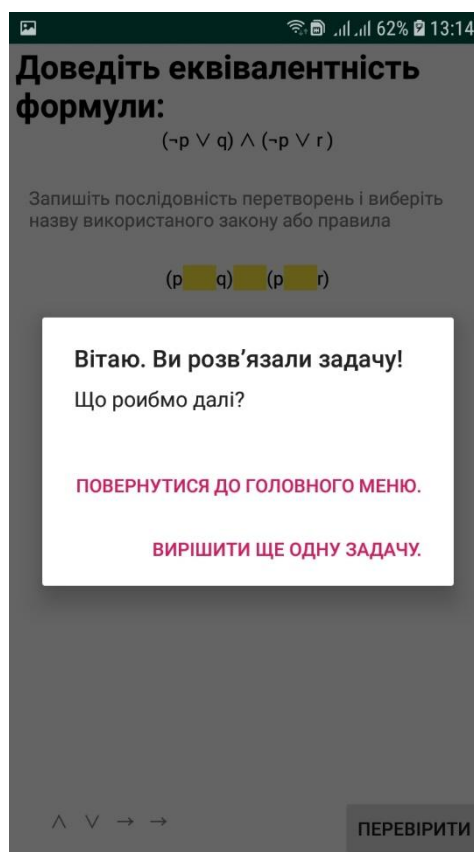


Рисунок 4.24. Перехід до головного меню після розв'язку останнього кроку задачі

4.4. Необхідна користувачу програми інструкція

Для користування даним тренажером користувачу потрібно мати смартфон з android 7.0 і вище. Завантажити тренажер можна буде з сайту дистанційної освіти в відповідному розділі. Після встановлення програми, користувач може приступати до вивчення теми, яку пропонує тренажер. Тренажер має інтуїтивно зрозумілий інтерфейс, тому проблем з його використання виникнути не повинно.

ВИСНОВКИ

В ході виконання магістерської роботи було проаналізовано літературні джерала та програми-тренажери. Була виявлена відсутність тренажерів на дану тему. Була досягнута поставлена мета магістерської роботи — створений тренажер, який буде використовуватись в навчанні студентів розв'язувати задачі з використання логіки висловлювань, закріплення знань та навичок. При цьому були здобуті та закріплені навички в програмуванні навчального тренажеру.

Розроблений алгоритм роботи та блок схема тренажеру. Були протестовані всі кроки тренажера на відсутність помилок та багів.

Розроблений тренажер буде впроваджено в дистанційний курс «Математична логіка» в ПУЕТ.

Додаток А — компакт-диск з програмою-тренажером.

Додаток Б — код програми-тренажера з теми «Логіка висловлювань» (типові кроки).

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Нікольський Ю.В., Пасічник В.В., Щербина Ю.М. Дискретна математика. — К.: Видавнича група BHV, 2007. — 368 с.
2. Математична логіка та теорія алгоритмів: практикум / уклад.: П. Ф. Жук – К. : НАУ, 2014. – 21 с.
3. Java 8. Карманный справочник / Р. Лигуори, П. Лигуори., 2017. – 256 с. – (ISBN: 9785845920508).
4. Head First Android Development / Jonathan Simon., 2011.-532с.- (ISBN: 978-1-449-39330-4).
5. Інформатика та системні науки (ІСН-2016) : матеріали VII Всеукраїнської науково-практичної конференції за міжнародною участю (м. Полтава, 10–12 березня 2016 року) / за ред. О.О. Ємця. - Полтава: ПУЕТ, 2016. - 362 с. Режим доступу - <http://dspace.puet.edu.ua/handle/123456789/3676>
6. Інформатика та системні науки (ІСН – 2017) : матеріали VIII Всеукраїнської науково-практичної конференції за міжнародною участю (м. Полтава, 16–18 березня 2017 р.) / за ред. Ємця О. О. – Полтава : ПУЕТ, 2017. – 333 с. Режим доступу - <http://dspace.puet.edu.ua/handle/123456789/5798>.
7. КОМП’ЮТЕРНІ НАУКИ І ПРИКЛАДНА МАТЕМАТИКА (КНіПМ-2018): матеріали науково-практичного семінару. Випуск 1 / за ред. Ємця О.О. – Полтава: Кафедра ММСІ ПУЕТ, 2018. – 64 с. Режим доступу - <http://dspace.puet.edu.ua/handle/123456789/6563>.
8. КОМП’ЮТЕРНІ НАУКИ І ПРИКЛАДНА МАТЕМАТИКА (КНіПМ-2018): матеріали науково-практичного семінару. Випуск 2 / за ред. Ємця О.О. – Полтава: Кафедра ММСІ ПУЕТ, 2018. – 27 с. Режим доступу - <http://dspace.puet.edu.ua/handle/123456789/6987>
9. Кондрашев Д. М. Розробка тренажера дистанційного навчального курсу «Математична логіка та теорія алгоритмів» з теми «Машини Т’юрінга» / Д. М. Кондрашев // Інформатика та системні науки (ІСН – 2017) :

- матеріали VIII Всеукраїнської науково-практичної конференції за міжнародною участю (м. Полтава, 16–18 березня 2017 р.) / за ред. Ємця О. О. – Полтава : ПУЕТ, 2017. — С. 146-149. Режим доступу: <http://dspace.puet.edu.ua/handle/123456789/5798>.
10. Потерайло О. О. Програмування навчального тренажера для градієнтного методу оптимізації нелінійних функцій дистанційного навчального курсу «Методи оптимізації та дослідження операцій» / О. О. Потерайло, О. О. Ємець // Інформатика та системні науки (ІСН – 2017) : матеріали VIII Всеукраїнської науково-практичної конференції за міжнародною участю (м. Полтава, 16–18 березня 2017 р.) / за ред. Ємця О. О. – Полтава : ПУЕТ, 2017. — С. 223-225. Режим доступу: <http://dspace.puet.edu.ua/handle/123456789/5798>.
 11. Русін В. С. Програмна реалізація елементів тренажеру з теми "Аналіз алгоритму сортування вставками" дисципліни "Аналіз алгоритмів" / В. С. Русін, Ю. Ф. Олексійчук // Інформатика та системні науки (ІСН-2017): матеріали VIII Всеукраїнської науково-практичної конференції за міжнародною участю (м. Полтава, 16–18 березня 2017 р.) – Полтава: ПУЕТ, 2017. – С. 236-237.
 12. Рудяга Д.С. Програмне забезпечення в методі аналізу ієрархій для системного аналізу при виборі альтернативних рішень в «ПП Ткаченко А.С.» / Д. С. Рудяга // КОМП'ЮТЕРНІ НАУКИ І ПРИКЛАДНА МАТЕМАТИКА (КНіПМ-2018): матеріали науково-практичного семінару. Випуск 1 / за ред. Ємця О.О. – Полтава: Кафедра ММСІ ПУЕТ, 2018. — 45 - 54 с. Режим доступу - <http://dspace.puet.edu.ua/handle/123456789/6498>
 13. Стовбун Д. О. Алгоритмізація та програмування елементів тренажера з побудови математичної моделі комбінаторної оптимізаційної задачі про оптимізацію суміші / Д. О. Стовбун // КОМП'ЮТЕРНІ НАУКИ І ПРИКЛАДНА МАТЕМАТИКА (КНіПМ-2018): матеріали науково-практичного семінару. Випуск 1 / за ред. Ємця О.О. – Полтава: Кафедра

- ММСІ ПУЕТ, 2018. – 45 - 54 с. Режим доступу - <http://dspace.puet.edu.ua/handle/123456789/6485>
14. Безменов Е. Г. Розробка програмного забезпечення для тренажера з теми «Неоднорідні диференціальні рівняння другого порядку із сталими коефіцієнтами» з дисципліни «Математичний аналіз» / Е. Г. Безменов // КОМП'ЮТЕРНІ НАУКИ І ПРИКЛАДНА МАТЕМАТИКА (КНіПМ-2018): матеріали науково-практичного семінару. Випуск 1 / за ред. Ємця О.О. – Полтава: Кафедра ММСІ ПУЕТ, 2018. – 45 - 54 с. Режим доступу - <http://dspace.puet.edu.ua/handle/123456789/6979>
 15. Bah Abibu, Parfonova T. O. Algorithm of the simulator on the topic “a straight line in space” / Bah Abibu, T. O. Parfonova // КОМП'ЮТЕРНІ НАУКИ І ПРИКЛАДНА МАТЕМАТИКА (КНіПМ-2018): матеріали науково-практичного семінару. Випуск 1 / за ред. Ємця О.О. – Полтава: Кафедра ММСІ ПУЕТ, 2018. – 45 - 54 с. Режим доступу - <http://dspace.puet.edu.ua/handle/123456789/6484>
 16. Кильник В. В. Алгоритм тренажера з теми «Сортування методом перемішування» / В. В. Кильник // Інформатика та системні науки (ІСН – 2017) : матеріали VIII Всеукраїнської науково-практичної конференції за міжнародною участю (м. Полтава, 16–18 березня 2017 р.) / за ред. Ємця О. О. – Полтава : ПУЕТ, 2017. — С. 135-137. Режим доступу: <http://dspace.puet.edu.ua/handle/123456789/5798>.
 17. Марченко Д. А. Програмування навчального тренажера з теми «Переставні многогранники / Д. А. Марченко, О. О. Ємець // Інформатика та системні науки (ІСН – 2017) : матеріали VIII Всеукраїнської науково-практичної конференції за міжнародною участю (м. Полтава, 16–18 березня 2017 р.) / за ред. Ємця О. О. – Полтава : ПУЕТ, 2017. — С. 190-192. Режим доступу: <http://dspace.puet.edu.ua/handle/123456789/5798>.
 18. Педоренко С. В. Розробка тренажера для М-методу в дистанційному курсі «Методи оптимізації та дослідження операцій» [Електронний ресурс] / С. В. Педоренко, О. О. Ємець // Інформатика та системні науки

- (ICN-2016) : матеріали VII Всеукр. наук.-практ. конф. за міжн. участю (м. Полтава, 10–12 берез. 2016 р.). – Полтава : ПУЕТ, 2016. – С. 226–230. – Режим доступу: <http://dspace.puet.edu.ua/handle/123456789/2946>. – Назва з екрана.
19. Сокол О. В. Розробка тренажера з теми «Нормальні алгоритми» дистанційного навчального курсу «Теорія алгоритмів» / О. В. Сокол, О. О. Черненко // Інформатика та системні науки (ICN – 2017) : матеріали VIII Всеукраїнської науково-практичної конференції за міжнародною участю (м. Полтава, 16–18 березня 2017 р.) / за ред. Ємця О. О. – Полтава: ПУЕТ, 2017. — С. 247-252. Режим доступу: <http://dspace.puet.edu.ua/handle/123456789/5798>.
 20. Глинський К.Г. Розробка тренажера з теми «Двоїстий симплекс метод» дистанційного навчального курсу / К.Г. Глинський // Інформатика та системні науки (ICN-2013) : матеріали IV Всеукр.наук.-практ. конф., (м. Полтава, 21–23 берез. 2013 р.). – Полтава: ПУЕТ, 2013. – С. 57-58. Режим доступу: <http://dspace.puet.edu.ua/handle/123456789/1628>.
 21. Койнаш А.М. Розробка програмного забезпечення тренажера з теми "Симплекс-метод" дистанційного навчального курсу "Методи оптимізації та дослідження операцій" / А.М. Койнаш // Інформатика та системні науки (ICN-2014) : матеріали V Всеукр.наук.-практ. конф., (м. Полтава, 13–15 березня 2014 р.). – Полтава: ПУЕТ, 2014. – С. 153-154. Режим доступу: <http://dspace.puet.edu.ua/handle/123456789/2833>.
 22. Овчаренко О.С. Алгоритмізація та програмне забезпечення тренажера з теми "Метод Жордана-Гауса" дистанційного навчального курсу "Методи оптимізації та дослідження операцій" / О.С. Овчаренко // Інформатика та системні науки (ICN-2014) : матеріали V Всеукр.наук.-практ. конф., (м. Полтава, 13–15 березня 2014 р.). – Полтава: ПУЕТ, 2014. – С. 231-233. Режим доступу: <http://dspace.puet.edu.ua/handle/123456789/1942>.
 23. Самовик С.М. Розробка алгоритму та програмного забезпечення тренажера з теми "Угорський метод в задачі про призначення"

- дистанційного навчального курсу "Методи оптимізації та дослідження операцій" / С.М. Самовик // Інформатика та системні науки (ІСН-2014) : матеріали V Всеукр.наук.-практ. конф., (м. Полтава, 13–15 березня 2014 р.). – Полтава: ПУЕТ, 2014. – С. 273-274. Режим доступу: <http://dspace.puet.edu.ua/handle/123456789/2840>.
- 24.Смирнов А.Б. Алгоритмізація та програмування тренажера з теми "Метод Форда-Фалкерсона" дистанційного навчального курсу "Методи оптимізації та дослідження операцій" / А.Б. Смирнов // Інформатика та системні науки (ІСН-2014) : матеріали V Всеукр.наук.-практ. конф., (м. Полтава, 13–15 березня 2014 р.). – Полтава: ПУЕТ, 2014. – С. 287-289. Режим доступу: <http://dspace.puet.edu.ua/handle/123456789/2841>.
- 25.Ємець О. О. Методичні рекомендації до виконання кваліфікаційної бакалаврської роботи для студентів 4 курсу напрямку підготовки 6.040302 "Інформатика" / О. О. Ємець, Ол-ра. О. Ємець. – Полтава: РВВ ПУЕТ, 2017. – 73 с.
- 26.Сивокінь О. Ю. Розробка та програмна реалізація тренажера з теми «Метод Ленда та Дойга» дистанційного курсу «Методи оптимізації та дослідження операцій» / О. Ю. Сивокінь, О. О. Ємець // Комп'ютерні науки і прикладна математика (КНіПМ-2018). – Полтава: ПУЕТ, 2018. Режим доступу: <http://dspace.puet.edu.ua/handle/123456789/5988>.
27. О. Y. Syvokin "The Simulator on the Topic "Land And Doig Method" of the Distance Learning Course "Optimization Methods and Operations Research": Development and Software Realization" / О. Y. Syvokin and О. О. Iemets // The VIth International Conference « Advanced Information Systems and Technologies, AIST 2018», (Sumy, 16-18 May 2018). – Sumy, Ukraine. 2018. – 72-75 с.

ДОДАТКИ

Додаток А Компакт-диск з тренажером

Додаток Б.1

Програмний код програми-тренажера (крок 1.1)

```

package com.e.diplomamaster.learning.lessons;

import android.content.Intent;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.CheckBox;
import android.widget.TextView;
import android.widget.Toast;

import com.e.diplomamaster.R;

public class FirstLesson extends AppCompatActivity {

    private Button check;
    private TextView changeExp, Title;
    private CheckBox ch1, ch2, ch3, ch4;
    Intent intent = null;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_first_lesson);

        ch1 = findViewById(R.id.ch1);
        ch2 = findViewById(R.id.ch2);
        ch3 = findViewById(R.id.ch3);
        ch4 = findViewById(R.id.ch4);
        Title = findViewById(R.id.textlogic);
        changeExp = findViewById(R.id.eplain1);
        check = findViewById(R.id.nextBut1);
        check.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                if(ch1.isChecked() && ch2.isChecked()){
                    intent = new Intent(FirstLesson.this, SecondLesson.class);
                    startActivity(intent);
                }
                if(ch3.isChecked() && ch4.isChecked()) {
                    Toast toast =
Toast.makeText(getApplicationContext(), R.string.promt_1, Toast.LENGTH_LONG);
                    toast.show();
                }
                if(ch3.isChecked()) {
                    Toast toast =
Toast.makeText(getApplicationContext(), R.string.promt_1, Toast.LENGTH_LONG);
                    toast.show();
                }
                if(ch4.isChecked()) {
                    Toast toast =
Toast.makeText(getApplicationContext(), R.string.promt_1, Toast.LENGTH_LONG);
                    toast.show();
                }
            }
        });
    }
}

```

```

<?xml version="1.0" encoding="utf-8" ?>

<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".learning.lessons.FirstLesson">

    <RelativeLayout
        android:id="@+id/upbar"
        android:layout_width="match_parent"
        android:layout_height="20dp"
        android:background="@color/upBar">

        <LinearLayout
            android:layout_width="wrap_content"
            android:layout_height="match_parent"
            android:layout_centerHorizontal="true">

            <FrameLayout
                android:layout_width="wrap_content"
                android:layout_height="match_parent"/>

            <FrameLayout
                android:layout_width="wrap_content"
                android:layout_height="match_parent"/>
        </LinearLayout>
    </RelativeLayout>

    <RelativeLayout
        android:id="@+id/qwest"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginHorizontal="25dp"
        android:layout_below="@+id/upbar"
        android:layout_marginTop="20dp">

        <TextView
            android:id="@+id/textlogic"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="@string/nametest1"
            android:textStyle="bold"
            android:textColor="#000000"
            android:layout_centerHorizontal="true"
            android:textSize="25sp"/>

        <Space
            android:id="@+id/space"
            android:layout_width="match_parent"
            android:layout_height="30dp"
            android:layout_below="@+id/textlogic"/>

        <TextView
            android:id="@+id/epain1"
            android:layout_below="@+id/space"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:textColor="#000000"
            android:text="@string/test1"
            />

        <Space
            android:id="@+id/spacel"
            android:layout_width="match_parent"

```

```

        android:layout_height="30dp"
        android:layout_below="@+id/explain1"/>

</RelativeLayout>

<RelativeLayout
    android:layout_below="@+id/qwest"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginHorizontal="25dp"
    android:layout_centerHorizontal="true">

    <CheckBox
        android:id="@+id/ch1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="@string/ch_1"
    />
    <CheckBox
        android:id="@+id/ch2"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="@string/ch_2"
        android:layout_toEndOf="@+id/ch1"
    />
    <CheckBox
        android:id="@+id/ch3"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="@string/ch_3"
        android:layout_toEndOf="@+id/ch2"
    />
    <CheckBox
        android:id="@+id/ch4"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="@string/ch_4"
        android:layout_toEndOf="@+id/ch3"
    />

</RelativeLayout>

<Button
    android:id="@+id/nextBut1"
    android:text="@string/check"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignParentBottom="true"
    android:layout_alignParentEnd="true"/>

</RelativeLayout>

```

Додаток Б.2

Програмний код програми-тренажера (крок 2.1.1)

```

package com.e.diplomamaster.learning.practice;

import android.content.ClipData;
import android.content.Intent;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.DragEvent;
import android.view.MotionEvent;
import android.view.View;
import android.widget.Button;
import android.widget.LinearLayout;
import android.widget.RadioButton;
import android.widget.RadioGroup;
import android.widget.TextView;
import android.widget.Toast;

import com.e.diplomamaster.R;
import com.e.diplomamaster.learning.lessons.SecondLesson;
import com.e.diplomamaster.learning.lessons.ThirdLesson;

public class PracticeLesson extends AppCompatActivity {

    LinearLayout targetBox1, targetBox2, targetBox3;
    private TextView txt1,txt2,txt3,txt4;
    private TextView box1, box2, box3;
    private Button check, ansWqwest;
    private Intent intent;
    private RadioButton ch1P, ch2P,ch3P;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_practice_lesson);

        final RadioGroup rGroup1 = findViewById(R.id.rGroup1);
        ch1P = findViewById(R.id.ch1P);
        ch2P = findViewById(R.id.ch2P);
        ch3P = findViewById(R.id.ch3P);

        targetBox1 = findViewById(R.id.target1);
        targetBox2 = findViewById(R.id.target2);
        targetBox3 = findViewById(R.id.target3);
        targetBox1.setOnDragListener(dragListener);
        targetBox2.setOnDragListener(dragListener);
        targetBox3.setOnDragListener(dragListener);

        txt1 = findViewById(R.id.and);
        txt2 = findViewById(R.id.or);
        txt3 = findViewById(R.id.if1);
        txt4 = findViewById(R.id.notP);
        txt1.setOnTouchListener(touchClickListener);
        txt2.setOnTouchListener(touchClickListener);
        txt3.setOnTouchListener(touchClickListener);
        txt4.setOnTouchListener(touchClickListener);

        box1 = findViewById(R.id.box11);
        box2 = findViewById(R.id.box22);
        box3 = findViewById(R.id.box33);
    }
}

```

```

    answQwest = findViewById(R.id.answTest1);
    answQwest.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            rGroup1.setVisibility(View.VISIBLE);
            answQwest.setVisibility(View.GONE);
        }
    });

    check = findViewById(R.id.nextBut2);
    check.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            if(ch1P.isChecked()){
                intent = new Intent(PracticeLesson.this,
PracticeLesson2.class);
                startActivity(intent);}else{
                    Toast toast = Toast.makeText(getApplicationContext(),
R.string.for_drag_elem2, Toast.LENGTH_SHORT);
                    toast.show();
                }
            }
        });
    }

    View.OnTouchListener touchClickListener = new View.OnTouchListener() {

        @Override
        public boolean onTouch(View v, MotionEvent event) {

            ClipData data = ClipData.newPlainText("", "");
            View.DragShadowBuilder myShadowBuilder = new
View.DragShadowBuilder(v);
            v.startDrag(data, myShadowBuilder,v,0);
            return true;
        }
    };

    View.OnDragListener dragListener = new View.OnDragListener(){
        @Override
        public boolean onDrag(View v, DragEvent event) {

            int dragEvent = event.getAction();
            final View view = (View) event.getLocalState();

            switch (dragEvent){
                case DragEvent.ACTION_DRAG_ENTERED:

                    break;
                case DragEvent.ACTION_DRAG_EXITED:
                    break;
                case DragEvent.ACTION_DROP:

                    if (view.getId()==R.id.if1 && v.getId()==R.id.target1){
                        LinearLayout oldPar = (LinearLayout) view.getParent();
                        oldPar.removeView(view);
                        LinearLayout newPar = (LinearLayout)v;
                        box1.setVisibility(View.GONE);
                        newPar.addView(view);

                    }else if(view.getId()==R.id.notP &&
v.getId()==R.id.target2){
                        LinearLayout oldPar = (LinearLayout) view.getParent();
                        oldPar.removeView(view);
                        LinearLayout newPar = (LinearLayout)v;

```



```

        box2.setVisibility(View.GONE);
        newPar.addView(view);

    }else if(view.getId()==R.id.and && v.getId()==R.id.target3){
        LinearLayout oldPar = (LinearLayout) view.getParent();
        oldPar.removeView(view);
        LinearLayout newPar = (LinearLayout)v;
        box3.setVisibility(View.GONE);
        newPar.addView(view);

    } else {
        Toast toast = Toast.makeText(getApplicationContext(),
R.string.for_drag_elem, Toast.LENGTH_SHORT);
        toast.show();
    }
    break;
}
return true;
}
};
}

```

```

<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".learning.practice.PracticeLesson">

    <RelativeLayout
        android:layout_width="fill_parent"
        android:layout_height="fill_parent"
        android:orientation="vertical">

        <TextView
            android:id="@+id/pract1Task"
            android:layout_width="350dp"
            android:layout_height="wrap_content"
            android:layout_centerHorizontal="true"
            android:text="@string/pract1"
            android:textColor="#000000"
            android:textSize="25sp"
            android:textStyle="bold" />

        <TextView
            android:id="@+id/pract1Expl"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_below="@+id/pract1Task"
            android:layout_centerHorizontal="true"
            android:text="@string/taskforPract"
            android:textColor="#000000" />

        <LinearLayout
            android:id="@+id/firstBox"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_below="@+id/pract1Expl"
            android:layout_marginStart="15dp"
            android:layout_marginTop="25dp"
            android:orientation="vertical">

```

```

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="@string/subtask1" />

    </LinearLayout>

    <Space
        android:id="@+id/firstSpace"
        android:layout_width="match_parent"
        android:layout_height="25dp"
        android:layout_below="@+id/firstBox" />

    <!-- початок формули в яку закидують елементи-->

    <LinearLayout
        android:id="@+id/afterEx"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_below="@+id/firstSpace"
        android:layout_centerHorizontal="true"
        android:orientation="horizontal">

        <LinearLayout
            android:id="@+id/subbox1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:orientation="vertical">

            <TextView
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:text="@string/varP"
                android:textColor="#000000" />

        </LinearLayout>

        <LinearLayout
            android:id="@+id/target1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:orientation="vertical">

            <TextView
                android:id="@+id/box11"
                android:layout_width="25dp"
                android:layout_height="match_parent"
                android:layout_gravity="center"
                android:background="#FFEB3B" />

        </LinearLayout>

        <LinearLayout
            android:id="@+id/subbox2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_gravity="start"
            android:orientation="vertical">

            <TextView
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:text="@string/qANDr"
                android:textColor="#000000" />

        </LinearLayout>

```

```

<LinearLayout
    android:id="@+id/target2"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:orientation="vertical">

    <TextView
        android:id="@+id/box22"
        android:layout_width="25dp"
        android:layout_height="match_parent"
        android:layout_gravity="center"
        android:background="#FFEB3B" />
</LinearLayout>

<LinearLayout
    android:id="@+id/subbox3"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_gravity="start"
    android:orientation="vertical">

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="@string/orQ"
        android:textColor="#000000" />
</LinearLayout>

<LinearLayout
    android:id="@+id/target3"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:orientation="vertical">

    <TextView
        android:id="@+id/box33"
        android:layout_width="25dp"
        android:layout_height="match_parent"
        android:layout_gravity="center"
        android:background="#FFEB3B" />
</LinearLayout>

<LinearLayout
    android:id="@+id/subbox4"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_gravity="start"
    android:orientation="vertical">

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="@string/rBracket"
        android:textColor="#000000" />
</LinearLayout>

</LinearLayout>

<!-- Кінець формули в яку закидують елементи-->

<LinearLayout
    android:id="@+id/target222"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"

```

```

        android:layout_alignParentBottom="true"
        android:layout_marginStart="25dp"
        android:layout_marginBottom="20dp"
        android:orientation="horizontal">

        <TextView
            android:id="@+id/and"
            android:layout_width="25dp"
            android:layout_height="15dp"
            android:gravity="center"
            android:text="@string/and" />

        <TextView
            android:id="@+id/or"
            android:layout_width="25dp"
            android:layout_height="15dp"
            android:gravity="center"
            android:text="@string/or" />

        <TextView
            android:id="@+id/if1"
            android:layout_width="25dp"
            android:layout_height="15dp"
            android:gravity="center"
            android:text="@string/arrowLeft" />

        <TextView
            android:id="@+id/notP"
            android:layout_width="25dp"
            android:layout_height="20dp"
            android:gravity="center"
            android:text="@string/varNotP" />

    </LinearLayout>

    <LinearLayout
        android:id="@+id/testAfter"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="110dp"
        android:orientation="vertical"
        android:layout_centerInParent="true">

        <RadioGroup
            android:id="@+id/rGroup1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_gravity="center"
            android:visibility="invisible">

            <RadioButton
                android:id="@+id/ch1P"
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:text="@string/ch1P" />

            <RadioButton
                android:id="@+id/ch2P"
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:text="@string/ch2P" />

            <RadioButton
                android:id="@+id/ch3P"

```

```

        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="@string/ch3P" />

    </RadioGroup>

</LinearLayout>

<Button
    android:id="@+id/answTest1"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="@string/answQwestion"
    android:layout_centerInParent="true"/>

<Button
    android:id="@+id/nextBut2"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignParentEnd="true"
    android:layout_alignParentBottom="true"
    android:text="@string/check"/>

</RelativeLayout>

</RelativeLayout>

```

Додаток Б.3

Програмний код програми-тренажера (крок 3.1.1)

```
package com.e.diplomamaster.learning.withExplain;

import android.content.Context;
import android.content.Intent;
import android.graphics.Color;
import android.graphics.drawable.AnimationDrawable;
import android.support.v7.app.AlertDialog;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.view.animation.Animation;
import android.view.animation.AnimationUtils;
import android.widget.Button;
import android.widget.ImageView;
import android.widget.LinearLayout;
import android.widget.TextView;

import com.e.diplomamaster.R;

public class ExerciseWithExplain1 extends AppCompatActivity {

    private Button solution;
    private TextView txtVWRed, explainForTable, solExplain, nextScreen,
falseRow53, falseRow54, falseRow55, falseRow56, falseRow46, trueRow45,
trueRow44, falseRow43, trueRow36, trueRow35, falseRow34, trueRow33, trueRow23,
falseRow23, trueRow25, trueRow24, falseRow24, falseRow25, falseRow26,
falseRow33, trueRow34, falseRow35, falseRow36, trueRow43, falseRow44,
falseRow45, trueRow46, trueRow53, trueRow54, trueRow55, trueRow56, trueRow26;
    LinearLayout layForTable;
    AlertDialog.Builder adForAnd, adForNot, adForIfelse;
    Context context;
    AnimationDrawable animationDrawable;
    ImageView animImage;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_exercise_with_explain1);
        trueRow23 = findViewById(R.id.trueRow23);
        falseRow23 = findViewById(R.id.falseRow23);
        falseRow24 = findViewById(R.id.falseRow24);
        trueRow24 = findViewById(R.id.trueRow24);
        trueRow25 = findViewById(R.id.trueRow25);
        falseRow25 = findViewById(R.id.falseRow25);
        trueRow26 = findViewById(R.id.trueRow26);
        falseRow26 = findViewById(R.id.falseRow26);
        falseRow33 = findViewById(R.id.falseRow33);
        trueRow33 = findViewById(R.id.trueRow33);
        trueRow34 = findViewById(R.id.trueRow34);
        falseRow34 = findViewById(R.id.falseRow34);
        falseRow35 = findViewById(R.id.falseRow35);
        trueRow35 = findViewById(R.id.trueRow35);
        falseRow36 = findViewById(R.id.falseRow36);
        trueRow36 = findViewById(R.id.trueRow36);
        trueRow43 = findViewById(R.id.trueRow43);
        falseRow44 = findViewById(R.id.falseRow44);
        falseRow45 = findViewById(R.id.falseRow45);
        falseRow43 = findViewById(R.id.falseRow43);
        trueRow44 = findViewById(R.id.trueRow44);
        trueRow45 = findViewById(R.id.trueRow45);
    }
}
```

```

trueRow46 = findViewById(R.id.trueRow46);
falseRow46 = findViewById(R.id.falseRow46);
trueRow53 = findViewById(R.id.trueRow53);
trueRow54 = findViewById(R.id.trueRow54);
trueRow55 = findViewById(R.id.trueRow55);
trueRow56 = findViewById(R.id.trueRow56);
falseRow53 = findViewById(R.id.falseRow53);
falseRow54 = findViewById(R.id.falseRow54);
falseRow55 = findViewById(R.id.falseRow55);
falseRow56 = findViewById(R.id.falseRow56);

animImage = findViewById(R.id.animImage);
animImage.setBackgroundResource(R.drawable.question_mark);
animationDrawable = (AnimationDrawable) animImage.getBackground();

context = ExerciseWithExplain1.this;
adForAnd = new AlertDialog.Builder(context);
adForAnd.setTitle(R.string.explainLogic);
adForAnd.setMessage(R.string.logicForAnd);

adForNot = new AlertDialog.Builder(context);
adForNot.setTitle(R.string.explainLogic);
adForNot.setMessage(R.string.logicForNot);

adForIfelse = new AlertDialog.Builder(context);
adForIfelse.setTitle(R.string.explainLogic);
adForIfelse.setMessage(R.string.logicForIfelse);

falseRow56.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        final CustomAnimationDrawableNew cad = new
CustomAnimationDrawableNew(
            (AnimationDrawable) getResources().getDrawable(
                R.drawable.question_mark)) {
            @Override
            public void onAnimationStart() {
                // Animation has started...
                animImage.setVisibility(View.VISIBLE);
                animationDrawable.start();
            }

            @Override
            public void onAnimationFinish() {
                // Animation has finished...
                animationDrawable.stop();
                animImage.setVisibility(View.GONE);
                adForNot.show();
            }
        };
        // Set the views drawable to our custom drawable
        //v.setBackgroundDrawable(cad);

        // Start the animation
        cad.start();
    }
});

trueRow56.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        final CustomAnimationDrawableNew cad = new
CustomAnimationDrawableNew(
            (AnimationDrawable) getResources().getDrawable(
                R.drawable.question_mark)) {

```

```

        @Override
        public void onAnimationStart() {
            // Animation has started...
            animImage.setVisibility(View.VISIBLE);
            animationDrawable.start();
        }

        @Override
        public void onAnimationFinish() {
            // Animation has finished...
            animationDrawable.stop();
            animImage.setVisibility(View.GONE);
            adForNot.show();
        }
    };
    // Set the views drawable to our custom drawable
    //v.setBackgroundDrawable(cad);

    // Start the animation
    cad.start();
}
});

falseRow55.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        final CustomAnimationDrawableNew cad = new
CustomAnimationDrawableNew(
            (AnimationDrawable) getResources().getDrawable(
                R.drawable.question_mark)) {
            @Override
            public void onAnimationStart() {
                // Animation has started...
                animImage.setVisibility(View.VISIBLE);
                animationDrawable.start();
            }

            @Override
            public void onAnimationFinish() {
                // Animation has finished...
                animationDrawable.stop();
                animImage.setVisibility(View.GONE);
                adForAnd.show();
            }
        };
        // Set the views drawable to our custom drawable
        //v.setBackgroundDrawable(cad);

        // Start the animation
        cad.start();
    }
});

trueRow55.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        final CustomAnimationDrawableNew cad = new
CustomAnimationDrawableNew(
            (AnimationDrawable) getResources().getDrawable(
                R.drawable.question_mark)) {
            @Override
            public void onAnimationStart() {
                // Animation has started...
                animImage.setVisibility(View.VISIBLE);
                animationDrawable.start();
            }

```



```

    }

    @Override
    public void onAnimationFinish() {
        // Animation has finished...
        animationDrawable.stop();
        animImage.setVisibility(View.GONE);
        adForAnd.show();
    }
};
// Set the views drawable to our custom drawable
//v.setBackgroundDrawable(cad);

// Start the animation
cad.start();
}
});

falseRow54.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        final CustomAnimationDrawableNew cad = new
CustomAnimationDrawableNew(
            (AnimationDrawable) getResources().getDrawable(
                R.drawable.question_mark)) {
            @Override
            public void onAnimationStart() {
                // Animation has started...
                animImage.setVisibility(View.VISIBLE);
                animationDrawable.start();
            }

            @Override
            public void onAnimationFinish() {
                // Animation has finished...
                animationDrawable.stop();
                animImage.setVisibility(View.GONE);
                adForNot.show();
            }
        };
        // Set the views drawable to our custom drawable
        //v.setBackgroundDrawable(cad);

        // Start the animation
        cad.start();
    }
});

trueRow54.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        final CustomAnimationDrawableNew cad = new
CustomAnimationDrawableNew(
            (AnimationDrawable) getResources().getDrawable(
                R.drawable.question_mark)) {
            @Override
            public void onAnimationStart() {
                // Animation has started...
                animImage.setVisibility(View.VISIBLE);
                animationDrawable.start();
            }

            @Override
            public void onAnimationFinish() {
                // Animation has finished...

```

```

        animationDrawable.stop();
        animImage.setVisibility(View.GONE);
        adForNot.show();
    }
};
// Set the views drawable to our custom drawable
//v.setBackgroundDrawable(cad);

// Start the animation
cad.start();
}
});

falseRow53.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        final CustomAnimationDrawableNew cad = new
CustomAnimationDrawableNew(
            (AnimationDrawable) getResources().getDrawable(
                R.drawable.question_mark)) {
            @Override
            public void onAnimationStart() {
                // Animation has started...
                animImage.setVisibility(View.VISIBLE);
                animationDrawable.start();
            }

            @Override
            public void onAnimationFinish() {
                // Animation has finished...
                animationDrawable.stop();
                animImage.setVisibility(View.GONE);
                adForIfelse.show();
            }
        };
        // Set the views drawable to our custom drawable
        //v.setBackgroundDrawable(cad);

        // Start the animation
        cad.start();
    }
});

trueRow53.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        final CustomAnimationDrawableNew cad = new
CustomAnimationDrawableNew(
            (AnimationDrawable) getResources().getDrawable(
                R.drawable.question_mark)) {
            @Override
            public void onAnimationStart() {
                // Animation has started...
                animImage.setVisibility(View.VISIBLE);
                animationDrawable.start();
            }

            @Override
            public void onAnimationFinish() {
                // Animation has finished...
                animationDrawable.stop();
                animImage.setVisibility(View.GONE);
                adForIfelse.show();
            }
        };
    }
});

```

```

        // Set the views drawable to our custom drawable
        //v.setBackgroundDrawable(cad);

        // Start the animation
        cad.start();
    }
});

falseRow46.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        final CustomAnimationDrawableNew cad = new
CustomAnimationDrawableNew(
            (AnimationDrawable) getResources().getDrawable(
                R.drawable.question_mark)) {
            @Override
            public void onAnimationStart() {
                // Animation has started...
                animImage.setVisibility(View.VISIBLE);
                animationDrawable.start();
            }

            @Override
            public void onAnimationFinish() {
                // Animation has finished...
                animationDrawable.stop();
                animImage.setVisibility(View.GONE);
                adForNot.show();
            }
        };
        // Set the views drawable to our custom drawable
        //v.setBackgroundDrawable(cad);

        // Start the animation
        cad.start();
    }
});

trueRow46.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        final CustomAnimationDrawableNew cad = new
CustomAnimationDrawableNew(
            (AnimationDrawable) getResources().getDrawable(
                R.drawable.question_mark)) {
            @Override
            public void onAnimationStart() {
                // Animation has started...
                animImage.setVisibility(View.VISIBLE);
                animationDrawable.start();
            }

            @Override
            public void onAnimationFinish() {
                // Animation has finished...
                animationDrawable.stop();
                animImage.setVisibility(View.GONE);
                adForNot.show();
            }
        };
        // Set the views drawable to our custom drawable
        //v.setBackgroundDrawable(cad);

        // Start the animation
        cad.start();
    }
});

```

```

    }
});

falseRow45.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        final CustomAnimationDrawableNew cad = new
CustomAnimationDrawableNew(
            (AnimationDrawable) getResources().getDrawable(
                R.drawable.question_mark)) {
            @Override
            public void onAnimationStart() {
                // Animation has started...
                animImage.setVisibility(View.VISIBLE);
                animationDrawable.start();
            }

            @Override
            public void onAnimationFinish() {
                // Animation has finished...
                animationDrawable.stop();
                animImage.setVisibility(View.GONE);
                adForAnd.show();
            }
        };
        // Set the views drawable to our custom drawable
        //v.setBackgroundDrawable(cad);

        // Start the animation
        cad.start();
    }
});

trueRow45.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        final CustomAnimationDrawableNew cad = new
CustomAnimationDrawableNew(
            (AnimationDrawable) getResources().getDrawable(
                R.drawable.question_mark)) {
            @Override
            public void onAnimationStart() {
                // Animation has started...
                animImage.setVisibility(View.VISIBLE);
                animationDrawable.start();
            }

            @Override
            public void onAnimationFinish() {
                // Animation has finished...
                animationDrawable.stop();
                animImage.setVisibility(View.GONE);
                adForAnd.show();
            }
        };
        // Set the views drawable to our custom drawable
        //v.setBackgroundDrawable(cad);

        // Start the animation
        cad.start();
    }
});

falseRow44.setOnClickListener(new View.OnClickListener() {
    @Override

```

```

        public void onClick(View v) {
            final CustomAnimationDrawableNew cad = new
CustomAnimationDrawableNew(
                (AnimationDrawable) getResources().getDrawable(
                    R.drawable.question_mark)) {
                @Override
                public void onAnimationStart() {
                    // Animation has started...
                    animImage.setVisibility(View.VISIBLE);
                    animationDrawable.start();
                }

                @Override
                public void onAnimationFinish() {
                    // Animation has finished...
                    animationDrawable.stop();
                    animImage.setVisibility(View.GONE);
                    adForNot.show();
                }
            };
            // Set the views drawable to our custom drawable
            //v.setBackgroundDrawable(cad);

            // Start the animation
            cad.start();
        }
    });

    trueRow44.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            final CustomAnimationDrawableNew cad = new
CustomAnimationDrawableNew(
                (AnimationDrawable) getResources().getDrawable(
                    R.drawable.question_mark)) {
                @Override
                public void onAnimationStart() {
                    // Animation has started...
                    animImage.setVisibility(View.VISIBLE);
                    animationDrawable.start();
                }

                @Override
                public void onAnimationFinish() {
                    // Animation has finished...
                    animationDrawable.stop();
                    animImage.setVisibility(View.GONE);
                    adForNot.show();
                }
            };
            // Set the views drawable to our custom drawable
            //v.setBackgroundDrawable(cad);

            // Start the animation
            cad.start();
        }
    });

    falseRow43.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            final CustomAnimationDrawableNew cad = new
CustomAnimationDrawableNew(
                (AnimationDrawable) getResources().getDrawable(
                    R.drawable.question_mark)) {

```

```

        @Override
        public void onAnimationStart() {
            // Animation has started...
            animImage.setVisibility(View.VISIBLE);
            animationDrawable.start();
        }

        @Override
        public void onAnimationFinish() {
            // Animation has finished...
            animationDrawable.stop();
            animImage.setVisibility(View.GONE);
            adForIfelse.show();
        }
    };
    // Set the views drawable to our custom drawable
    //v.setBackgroundDrawable(cad);

    // Start the animation
    cad.start();
}

});

trueRow43.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        final CustomAnimationDrawableNew cad = new
CustomAnimationDrawableNew(
            (AnimationDrawable) getResources().getDrawable(
                R.drawable.question_mark)) {
            @Override
            public void onAnimationStart() {
                // Animation has started...
                animImage.setVisibility(View.VISIBLE);
                animationDrawable.start();
            }

            @Override
            public void onAnimationFinish() {
                // Animation has finished...
                animationDrawable.stop();
                animImage.setVisibility(View.GONE);
                adForIfelse.show();
            }
        };
        // Set the views drawable to our custom drawable
        //v.setBackgroundDrawable(cad);

        // Start the animation
        cad.start();
    }
});

falseRow36.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        final CustomAnimationDrawableNew cad = new
CustomAnimationDrawableNew(
            (AnimationDrawable) getResources().getDrawable(
                R.drawable.question_mark)) {
            @Override
            public void onAnimationStart() {
                // Animation has started...
                animImage.setVisibility(View.VISIBLE);
                animationDrawable.start();
            }
        };
    }
});

```

```

    }

    @Override
    public void onAnimationFinish() {
        // Animation has finished...
        animationDrawable.stop();
        animImage.setVisibility(View.GONE);
        adForNot.show();
    }
};
// Set the views drawable to our custom drawable
//v.setBackgroundDrawable(cad);

// Start the animation
cad.start();
}
});

trueRow36.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        final CustomAnimationDrawableNew cad = new
CustomAnimationDrawableNew(
            (AnimationDrawable) getResources().getDrawable(
                R.drawable.question_mark)) {
            @Override
            public void onAnimationStart() {
                // Animation has started...
                animImage.setVisibility(View.VISIBLE);
                animationDrawable.start();
            }

            @Override
            public void onAnimationFinish() {
                // Animation has finished...
                animationDrawable.stop();
                animImage.setVisibility(View.GONE);
                adForNot.show();
            }
        };
        // Set the views drawable to our custom drawable
        //v.setBackgroundDrawable(cad);

        // Start the animation
        cad.start();
    }
});

falseRow35.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        final CustomAnimationDrawableNew cad = new
CustomAnimationDrawableNew(
            (AnimationDrawable) getResources().getDrawable(
                R.drawable.question_mark)) {
            @Override
            public void onAnimationStart() {
                // Animation has started...
                animImage.setVisibility(View.VISIBLE);
                animationDrawable.start();
            }

            @Override
            public void onAnimationFinish() {
                // Animation has finished...

```

```

        animationDrawable.stop();
        animImage.setVisibility(View.GONE);
        adForAnd.show();
    }
};
// Set the views drawable to our custom drawable
//v.setBackgroundDrawable(cad);

// Start the animation
cad.start();
}
});

trueRow35.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        final CustomAnimationDrawableNew cad = new
CustomAnimationDrawableNew(
            (AnimationDrawable) getResources().getDrawable(
                R.drawable.question_mark)) {
            @Override
            public void onAnimationStart() {
                // Animation has started...
                animImage.setVisibility(View.VISIBLE);
                animationDrawable.start();
            }

            @Override
            public void onAnimationFinish() {
                // Animation has finished...
                animationDrawable.stop();
                animImage.setVisibility(View.GONE);
                adForAnd.show();
            }
        };
        // Set the views drawable to our custom drawable
        //v.setBackgroundDrawable(cad);

        // Start the animation
        cad.start();
    }
});

falseRow34.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        final CustomAnimationDrawableNew cad = new
CustomAnimationDrawableNew(
            (AnimationDrawable) getResources().getDrawable(
                R.drawable.question_mark)) {
            @Override
            public void onAnimationStart() {
                // Animation has started...
                animImage.setVisibility(View.VISIBLE);
                animationDrawable.start();
            }

            @Override
            public void onAnimationFinish() {
                // Animation has finished...
                animationDrawable.stop();
                animImage.setVisibility(View.GONE);
                adForNot.show();
            }
        };
    }
};

```



```

        // Set the views drawable to our custom drawable
        //v.setBackgroundDrawable(cad);

        // Start the animation
        cad.start();
    }
});

trueRow34.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        final CustomAnimationDrawableNew cad = new
CustomAnimationDrawableNew(
            (AnimationDrawable) getResources().getDrawable(
                R.drawable.question_mark)) {
            @Override
            public void onAnimationStart() {
                // Animation has started...
                animImage.setVisibility(View.VISIBLE);
                animationDrawable.start();
            }

            @Override
            public void onAnimationFinish() {
                // Animation has finished...
                animationDrawable.stop();
                animImage.setVisibility(View.GONE);
                adForNot.show();
            }
        };
        // Set the views drawable to our custom drawable
        //v.setBackgroundDrawable(cad);

        // Start the animation
        cad.start();
    }
});

falseRow33.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        final CustomAnimationDrawableNew cad = new
CustomAnimationDrawableNew(
            (AnimationDrawable) getResources().getDrawable(
                R.drawable.question_mark)) {
            @Override
            public void onAnimationStart() {
                // Animation has started...
                animImage.setVisibility(View.VISIBLE);
                animationDrawable.start();
            }

            @Override
            public void onAnimationFinish() {
                // Animation has finished...
                animationDrawable.stop();
                animImage.setVisibility(View.GONE);
                adForIfelse.show();
            }
        };
        // Set the views drawable to our custom drawable
        //v.setBackgroundDrawable(cad);

        // Start the animation
        cad.start();
    }
});

```

```

    }
});

trueRow33.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        final CustomAnimationDrawableNew cad = new
CustomAnimationDrawableNew(
            (AnimationDrawable) getResources().getDrawable(
                R.drawable.question_mark)) {
            @Override
            public void onAnimationStart() {
                // Animation has started...
                animImage.setVisibility(View.VISIBLE);
                animationDrawable.start();
            }

            @Override
            public void onAnimationFinish() {
                // Animation has finished...
                animationDrawable.stop();
                animImage.setVisibility(View.GONE);
                adForIfelse.show();
            }
        };
        // Set the views drawable to our custom drawable
        //v.setBackgroundDrawable(cad);

        // Start the animation
        cad.start();
    }
});

falseRow26.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        final CustomAnimationDrawableNew cad = new
CustomAnimationDrawableNew(
            (AnimationDrawable) getResources().getDrawable(
                R.drawable.question_mark)) {
            @Override
            public void onAnimationStart() {
                // Animation has started...
                animImage.setVisibility(View.VISIBLE);
                animationDrawable.start();
            }

            @Override
            public void onAnimationFinish() {
                // Animation has finished...
                animationDrawable.stop();
                animImage.setVisibility(View.GONE);
                adForNot.show();
            }
        };
        // Set the views drawable to our custom drawable
        //v.setBackgroundDrawable(cad);

        // Start the animation
        cad.start();
    }
});

trueRow26.setOnClickListener(new View.OnClickListener() {
    @Override

```

```

        public void onClick(View v) {
            final CustomAnimationDrawableNew cad = new
CustomAnimationDrawableNew(
                (AnimationDrawable) getResources().getDrawable(
                    R.drawable.question_mark)) {
                @Override
                public void onAnimationStart() {
                    // Animation has started...
                    animImage.setVisibility(View.VISIBLE);
                    animationDrawable.start();
                }

                @Override
                public void onAnimationFinish() {
                    // Animation has finished...
                    animationDrawable.stop();
                    animImage.setVisibility(View.GONE);
                    adForNot.show();
                }
            };
            // Set the views drawable to our custom drawable
            //v.setBackgroundDrawable(cad);

            // Start the animation
            cad.start();
        }
    });

    falseRow25.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            final CustomAnimationDrawableNew cad = new
CustomAnimationDrawableNew(
                (AnimationDrawable) getResources().getDrawable(
                    R.drawable.question_mark)) {
                @Override
                public void onAnimationStart() {
                    // Animation has started...
                    animImage.setVisibility(View.VISIBLE);
                    animationDrawable.start();
                }

                @Override
                public void onAnimationFinish() {
                    // Animation has finished...
                    animationDrawable.stop();
                    animImage.setVisibility(View.GONE);
                    adForAnd.show();
                }
            };
            // Set the views drawable to our custom drawable
            //v.setBackgroundDrawable(cad);

            // Start the animation
            cad.start();
        }
    });

    trueRow25.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            final CustomAnimationDrawableNew cad = new
CustomAnimationDrawableNew(
                (AnimationDrawable) getResources().getDrawable(
                    R.drawable.question_mark)) {

```

```

        @Override
        public void onAnimationStart() {
            // Animation has started...
            animImage.setVisibility(View.VISIBLE);
            animationDrawable.start();
        }

        @Override
        public void onAnimationFinish() {
            // Animation has finished...
            animationDrawable.stop();
            animImage.setVisibility(View.GONE);
            adForAnd.show();
        }
    };
    // Set the views drawable to our custom drawable
    //v.setBackgroundDrawable(cad);

    // Start the animation
    cad.start();
}
});

falseRow24.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        final CustomAnimationDrawableNew cad = new
CustomAnimationDrawableNew(
            (AnimationDrawable) getResources().getDrawable(
                R.drawable.question_mark)) {
            @Override
            public void onAnimationStart() {
                // Animation has started...
                animImage.setVisibility(View.VISIBLE);
                animationDrawable.start();
            }

            @Override
            public void onAnimationFinish() {
                // Animation has finished...
                animationDrawable.stop();
                animImage.setVisibility(View.GONE);
                adForNot.show();
            }
        };
        // Set the views drawable to our custom drawable
        //v.setBackgroundDrawable(cad);

        // Start the animation
        cad.start();
    }
});

trueRow24.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        final CustomAnimationDrawableNew cad = new
CustomAnimationDrawableNew(
            (AnimationDrawable) getResources().getDrawable(
                R.drawable.question_mark)) {
            @Override
            public void onAnimationStart() {
                // Animation has started...
                animImage.setVisibility(View.VISIBLE);
                animationDrawable.start();
            }

```

```

    }

    @Override
    public void onAnimationFinish() {
        // Animation has finished...
        animationDrawable.stop();
        animImage.setVisibility(View.GONE);
        adForNot.show();
    }
};
// Set the views drawable to our custom drawable
//v.setBackgroundDrawable(cad);

// Start the animation
cad.start();
}
});

falseRow23.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        final CustomAnimationDrawableNew cad = new
CustomAnimationDrawableNew(
            (AnimationDrawable) getResources().getDrawable(
                R.drawable.question_mark)) {
            @Override
            public void onAnimationStart() {
                // Animation has started...
                animImage.setVisibility(View.VISIBLE);
                animationDrawable.start();
            }

            @Override
            public void onAnimationFinish() {
                // Animation has finished...
                animationDrawable.stop();
                animImage.setVisibility(View.GONE);
                adForIfelse.show();
            }
        };
        // Set the views drawable to our custom drawable
        //v.setBackgroundDrawable(cad);

        // Start the animation
        cad.start();
    }
});

trueRow23.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        final CustomAnimationDrawableNew cad = new
CustomAnimationDrawableNew(
            (AnimationDrawable) getResources().getDrawable(
                R.drawable.question_mark)) {
            @Override
            public void onAnimationStart() {
                // Animation has started...
                animImage.setVisibility(View.VISIBLE);
                animationDrawable.start();
            }

            @Override
            public void onAnimationFinish() {
                // Animation has finished...

```

```

        animationDrawable.stop();
        animImage.setVisibility(View.GONE);
        adForIfelse.show();
    }
};
// Set the views drawable to our custom drawable
//v.setBackgroundDrawable(cad);

// Start the animation
cad.start();
}
});

nextScreen = findViewById(R.id.nextStepExpl1);
layForTable = findViewById(R.id.layForTable3);

explainForTable = findViewById(R.id.explainForTable);
solExplain = findViewById(R.id.solExplain);
solution = findViewById(R.id.solution);

solution.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        solution.setVisibility(View.GONE);
        solExplain.setVisibility(View.VISIBLE);
        nextScreen.setVisibility(View.VISIBLE);
        layForTable.setVisibility(View.VISIBLE);
        startAnimation();
    }
});

nextScreen.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        Intent intent = new Intent(ExerciseWithExplain1.this,
ExerciseWithExplain1_2.class);
        startActivity(intent);
    }
});
}

private void startAnimation(){
    Animation animation = AnimationUtils.loadAnimation(this, R.anim.myalfa);
    solExplain.startAnimation(animation);
    layForTable.startAnimation(animation);
    Animation animation1 = AnimationUtils.loadAnimation(this,
R.anim.zoomin);
    explainForTable.startAnimation(animation1);

    int[] txtViewGreen = new int[]{R.id.trueRow23, R.id.falseRow24,
R.id.falseRow25, R.id.falseRow26, R.id.falseRow33, R.id.trueRow34,
R.id.falseRow35, R.id.falseRow36, R.id.trueRow43, R.id.falseRow44,
R.id.falseRow45, R.id.trueRow46, R.id.trueRow53, R.id.trueRow54, R.id.trueRow55,
R.id.trueRow56};
    for (int i1 : txtViewGreen) {
        TextView txtVWGreen = findViewById(i1);
        txtVWGreen.startAnimation(animation1);
        txtVWGreen.setBackgroundColor(Color.GREEN);
        int[] txtViewRed = new int[]{R.id.falseRow23, R.id.trueRow24,
R.id.trueRow25, R.id.trueRow26, R.id.trueRow33, R.id.falseRow34, R.id.trueRow35,
R.id.trueRow36, R.id.falseRow43, R.id.trueRow44, R.id.trueRow45,
R.id.falseRow46, R.id.falseRow53, R.id.falseRow54, R.id.falseRow55,
R.id.falseRow56};
        for (int i2 : txtViewRed) {
            txtVWRed = findViewById(i2);

```

```

        txtVWRed.startAnimation(animation1);
        txtVWRed.setBackgroundColor(Color.RED);
    }
}
}

<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".learning.withExplain.ExerciseWithExplain1">

    <LinearLayout
        android:id="@+id/firstLay"
        android:layout_alignParentTop="true"
        android:layout_marginTop="10dp"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="vertical"
        android:layout_centerHorizontal="true"
        android:layout_marginStart="25dp">

        <TextView
            android:id="@+id/firstHeader"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="@string/firstHeader"
            android:gravity="center"
            android:textColor="#000"
            android:textSize="20sp"
            android:textStyle="bold"/>

        <Space
            android:layout_width="match_parent"
            android:layout_height="20dp" />

        <TextView
            android:id="@+id/firstTaskForExp"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="@string/firstTaskForExp"
            android:gravity="center"
            android:textColor="#000"/>

    </LinearLayout>
    <Space
        android:id="@+id/spaceFirst"
        android:layout_width="match_parent"
        android:layout_height="20dp"
        android:layout_below="@+id/firstLay"/>

    <LinearLayout
        android:id="@+id/butSol"
        android:layout_below="@+id/spaceFirst"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_centerHorizontal="true"
        android:orientation="vertical">
        <Button
            android:id="@+id/solution"
            android:layout_width="wrap_content"

```

```

        android:layout_height="wrap_content"
        android:text="@string/solutionBut"
    />
</LinearLayout>

<LinearLayout
    android:id="@+id/solExp1"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_below="@+id/spaceFirst"
    android:layout_centerHorizontal="true"
    android:layout_marginStart="10dp">

    <TextView
        android:id="@+id/solExplain"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="@string/solExp1"
        android:gravity="center"
        android:visibility="invisible"
        android:textColor="#000"/>
</LinearLayout>

    <Space
        android:id="@+id/spaceSecond"
        android:layout_width="match_parent"
        android:layout_height="20dp"
        android:layout_below="@+id/solExp1"/>

    <LinearLayout
        android:id="@+id/layForTable3"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_below="@+id/spaceSecond"
        android:layout_centerHorizontal="true"
        android:background="#000"
        android:orientation="horizontal"
        android:visibility="invisible">

        <TableLayout
            android:layout_width="wrap_content"
            android:layout_height="wrap_content">

            <!--condition Table-->
            <TableRow android:layout_margin="1dp">

                <TextView
                    android:id="@+id/condition11"
                    android:layout_height="match_parent"
                    android:layout_weight="0"
                    android:background="#B9B9B9"
                    android:gravity="center"
                    android:padding="5dp"
                    android:text="@string/bigP"
                    android:textColor="#000"
                    android:textSize="15sp" />

                <TextView
                    android:id="@+id/condition12"
                    android:layout_marginStart="1dp"
                    android:layout_weight="0"
                    android:background="#B9B9B9"
                    android:gravity="center"
                    android:padding="5dp"
                    android:text="@string/bigQ"
                    android:textColor="#000"

```



```

        android:textSize="15sp" />

<TextView
    android:id="@+id/condition13"
    android:layout_marginStart="1dp"
    android:layout_weight="0"
    android:background="#B9B9B9"
    android:gravity="center"
    android:padding="5dp"
    android:text="@string/p_q"
    android:textColor="#000"
    android:textSize="15sp" />

<TextView
    android:id="@+id/condition14"
    android:layout_marginStart="1dp"
    android:layout_weight="0"
    android:background="#B9B9B9"
    android:gravity="center"
    android:padding="5dp"
    android:text="@string/notQ"
    android:textColor="#000"
    android:textSize="15sp" />

<TextView
    android:id="@+id/condition16"
    android:layout_marginStart="1dp"
    android:layout_weight="1"
    android:background="#B9B9B9"
    android:gravity="center"
    android:padding="5dp"
    android:text="@string/p_q_notQ"
    android:textColor="#000"
    android:textSize="15sp" />

<TextView
    android:id="@+id/condition17"
    android:layout_marginStart="1dp"
    android:layout_weight="0"
    android:background="#B9B9B9"
    android:gravity="center"
    android:padding="5dp"
    android:text="@string/notP"
    android:textColor="#000"
    android:textSize="15sp" />
</TableRow>

<!--true or false start1-->
<TableRow android:layout_margin="1dp">

    <TextView
        android:id="@+id/trueRow21"
        android:layout_height="match_parent"
        android:layout_weight="0"
        android:background="#FFFFFF"
        android:gravity="center"
        android:padding="5dp"
        android:text="@string/bigT"
        android:textColor="#000"
        android:textSize="15sp" />

    <TextView
        android:id="@+id/trueRow22"
        android:layout_marginStart="1dp"
        android:layout_weight="0"

```

```

        android:background="#FFFFFF"
        android:gravity="center"
        android:padding="5dp"
        android:text="@string/bigT"
        android:textColor="#000"
        android:textSize="15sp" />

```

```
<TableRow>
```

```

    <TextView
        android:id="@+id/trueRow23"
        android:layout_height="match_parent"
        android:layout_marginStart="1dp"
        android:layout_weight="1"
        android:background="#FFFFFF"
        android:gravity="center"
        android:padding="5dp"
        android:text="@string/bigT"
        android:textColor="#000"
        android:textSize="15sp" />

```

```

    <TextView
        android:id="@+id/falseRow23"
        android:layout_marginStart="1dp"
        android:layout_weight="1"
        android:background="#FFFFFF"
        android:gravity="center"
        android:padding="5dp"
        android:text="@string/bigf"
        android:textColor="#000"
        android:textSize="15sp" />

```

```
</TableRow>
```

```
<TableRow>
```

```

    <TextView
        android:id="@+id/trueRow24"
        android:layout_height="match_parent"
        android:layout_marginStart="1dp"
        android:layout_weight="1"
        android:background="#FFFFFF"
        android:gravity="center"
        android:padding="5dp"
        android:text="@string/bigT"
        android:textColor="#000"
        android:textSize="15sp" />

```

```

    <TextView
        android:id="@+id/falseRow24"
        android:layout_marginStart="1dp"
        android:layout_weight="1"
        android:background="#FFFFFF"
        android:gravity="center"
        android:padding="5dp"
        android:text="@string/bigf"
        android:textColor="#000"
        android:textSize="15sp" />

```

```
</TableRow>
```

```
<TableRow>
```

```

    <TextView
        android:id="@+id/trueRow25"
        android:layout_height="match_parent"

```

```

        android:layout_marginStart="1dp"
        android:layout_weight="1"
        android:background="#FFFFFF"
        android:gravity="center"
        android:padding="5dp"
        android:text="@string/biggt"
        android:textColor="#000"
        android:textSize="15sp" />

<TextView
    android:id="@+id/falseRow25"
    android:layout_marginStart="1dp"
    android:layout_weight="1"
    android:background="#FFFFFF"
    android:gravity="center"
    android:padding="5dp"
    android:text="@string/bigf"
    android:textColor="#000"
    android:textSize="15sp" />
</TableRow>

<TableRow>

    <TextView
        android:id="@+id/trueRow26"
        android:layout_height="match_parent"
        android:layout_marginStart="1dp"
        android:layout_weight="1"
        android:background="#FFFFFF"
        android:gravity="center"
        android:padding="5dp"
        android:text="@string/biggt"
        android:textColor="#000"
        android:textSize="15sp" />

    <TextView
        android:id="@+id/falseRow26"
        android:layout_marginStart="1dp"
        android:layout_weight="1"
        android:background="#FFFFFF"
        android:gravity="center"
        android:padding="5dp"
        android:text="@string/bigf"
        android:textColor="#000"
        android:textSize="15sp" />
</TableRow>
</TableRow>

<!--true or false start2-->
<TableRow android:layout_margin="1dp">

    <TextView
        android:id="@+id/trueRow31"
        android:layout_height="match_parent"
        android:layout_weight="0"
        android:background="#FFFFFF"
        android:gravity="center"
        android:padding="5dp"
        android:text="@string/biggt"
        android:textColor="#000"
        android:textSize="15sp" />

    <TextView
        android:id="@+id/falseRow32"
        android:layout_marginStart="1dp"

```

```

        android:layout_weight="0"
        android:background="#FFFFFF"
        android:gravity="center"
        android:padding="5dp"
        android:text="@string/bigf"
        android:textColor="#000"
        android:textSize="15sp" />

```

```
<TableRow>
```

```

    <TextView
        android:id="@+id/trueRow33"
        android:layout_height="match_parent"
        android:layout_marginStart="1dp"
        android:layout_weight="1"
        android:background="#FFFFFF"
        android:gravity="center"
        android:padding="5dp"
        android:text="@string/bigf"
        android:textColor="#000"
        android:textSize="15sp" />

```

```

    <TextView
        android:id="@+id/falseRow33"
        android:layout_marginStart="1dp"
        android:layout_weight="1"
        android:background="#FFFFFF"
        android:gravity="center"
        android:padding="5dp"
        android:text="@string/bigf"
        android:textColor="#000"
        android:textSize="15sp" />

```

```
</TableRow>
```

```
<TableRow>
```

```

    <TextView
        android:id="@+id/trueRow34"
        android:layout_height="match_parent"
        android:layout_marginStart="1dp"
        android:layout_weight="1"
        android:background="#FFFFFF"
        android:gravity="center"
        android:padding="5dp"
        android:text="@string/bigf"
        android:textColor="#000"
        android:textSize="15sp" />

```

```

    <TextView
        android:id="@+id/falseRow34"
        android:layout_marginStart="1dp"
        android:layout_weight="1"
        android:background="#FFFFFF"
        android:gravity="center"
        android:padding="5dp"
        android:text="@string/bigf"
        android:textColor="#000"
        android:textSize="15sp" />

```

```
</TableRow>
```

```
<TableRow>
```

```

    <TextView
        android:id="@+id/trueRow35"

```

```

        android:layout_height="match_parent"
        android:layout_marginStart="1dp"
        android:layout_weight="1"
        android:background="#FFFFFF"
        android:gravity="center"
        android:padding="5dp"
        android:text="@string/biggt"
        android:textColor="#000"
        android:textSize="15sp" />

<TextView
    android:id="@+id/falseRow35"
    android:layout_marginStart="1dp"
    android:layout_weight="1"
    android:background="#FFFFFF"
    android:gravity="center"
    android:padding="5dp"
    android:text="@string/bigf"
    android:textColor="#000"
    android:textSize="15sp" />
</TableRow>

<TableRow>

    <TextView
        android:id="@+id/trueRow36"
        android:layout_height="match_parent"
        android:layout_marginStart="1dp"
        android:layout_weight="1"
        android:background="#FFFFFF"
        android:gravity="center"
        android:padding="5dp"
        android:text="@string/biggt"
        android:textColor="#000"
        android:textSize="15sp" />

    <TextView
        android:id="@+id/falseRow36"
        android:layout_marginStart="1dp"
        android:layout_weight="1"
        android:background="#FFFFFF"
        android:gravity="center"
        android:padding="5dp"
        android:text="@string/bigf"
        android:textColor="#000"
        android:textSize="15sp" />
</TableRow>
</TableRow>

<!--true or false start3-->
<TableRow android:layout_margin="1dp">

    <TextView
        android:id="@+id/trueRow41"
        android:layout_height="match_parent"
        android:layout_weight="0"
        android:background="#FFFFFF"
        android:gravity="center"
        android:padding="5dp"
        android:text="@string/bigf"
        android:textColor="#000"
        android:textSize="15sp" />

    <TextView
        android:id="@+id/falseRow42"

```

```

        android:layout_marginStart="1dp"
        android:layout_weight="0"
        android:background="#FFFFFF"
        android:gravity="center"
        android:padding="5dp"
        android:text="@string/biggt"
        android:textColor="#000"
        android:textSize="15sp" />

```

```
<TableRow>
```

```

    <TextView
        android:id="@+id/trueRow43"
        android:layout_height="match_parent"
        android:layout_marginStart="1dp"
        android:layout_weight="1"
        android:background="#FFFFFF"
        android:gravity="center"
        android:padding="5dp"
        android:text="@string/biggt"
        android:textColor="#000"
        android:textSize="15sp" />

```

```

    <TextView
        android:id="@+id/falseRow43"
        android:layout_marginStart="1dp"
        android:layout_weight="1"
        android:background="#FFFFFF"
        android:gravity="center"
        android:padding="5dp"
        android:text="@string/bigf"
        android:textColor="#000"
        android:textSize="15sp" />

```

```
</TableRow>
```

```
<TableRow>
```

```

    <TextView
        android:id="@+id/trueRow44"
        android:layout_height="match_parent"
        android:layout_marginStart="1dp"
        android:layout_weight="1"
        android:background="#FFFFFF"
        android:gravity="center"
        android:padding="5dp"
        android:text="@string/biggt"
        android:textColor="#000"
        android:textSize="15sp" />

```

```

    <TextView
        android:id="@+id/falseRow44"
        android:layout_marginStart="1dp"
        android:layout_weight="1"
        android:background="#FFFFFF"
        android:gravity="center"
        android:padding="5dp"
        android:text="@string/bigf"
        android:textColor="#000"
        android:textSize="15sp" />

```

```
</TableRow>
```

```
<TableRow>
```

```

    <TextView

```

```

        android:id="@+id/trueRow45"
        android:layout_height="match_parent"
        android:layout_marginStart="1dp"
        android:layout_weight="1"
        android:background="#FFFFFF"
        android:gravity="center"
        android:padding="5dp"
        android:text="@string/biggt"
        android:textColor="#000"
        android:textSize="15sp" />

<TextView
    android:id="@+id/falseRow45"
    android:layout_marginStart="1dp"
    android:layout_weight="1"
    android:background="#FFFFFF"
    android:gravity="center"
    android:padding="5dp"
    android:text="@string/bigf"
    android:textColor="#000"
    android:textSize="15sp" />
</TableRow>

<TableRow>

    <TextView
        android:id="@+id/trueRow46"
        android:layout_height="match_parent"
        android:layout_marginStart="1dp"
        android:layout_weight="1"
        android:background="#FFFFFF"
        android:gravity="center"
        android:padding="5dp"
        android:text="@string/biggt"
        android:textColor="#000"
        android:textSize="15sp" />

    <TextView
        android:id="@+id/falseRow46"
        android:layout_marginStart="1dp"
        android:layout_weight="1"
        android:background="#FFFFFF"
        android:gravity="center"
        android:padding="5dp"
        android:text="@string/bigf"
        android:textColor="#000"
        android:textSize="15sp" />
</TableRow>
</TableRow>

<!--true or false start4-->
<TableRow android:layout_margin="1dp">

    <TextView
        android:id="@+id/trueRow51"
        android:layout_height="match_parent"
        android:layout_weight="0"
        android:background="#FFFFFF"
        android:gravity="center"
        android:padding="5dp"
        android:text="@string/bigf"
        android:textColor="#000"
        android:textSize="15sp" />

    <TextView

```

```

        android:id="@+id/falseRow52"
        android:layout_marginStart="1dp"
        android:layout_weight="0"
        android:background="#FFFFFF"
        android:gravity="center"
        android:padding="5dp"
        android:text="@string/bigf"
        android:textColor="#000"
        android:textSize="15sp" />

```

```
<TableRow>
```

```

    <TextView
        android:id="@+id/trueRow53"
        android:layout_height="match_parent"
        android:layout_marginStart="1dp"
        android:layout_weight="1"
        android:background="#FFFFFF"
        android:gravity="center"
        android:padding="5dp"
        android:text="@string/bigf"
        android:textColor="#000"
        android:textSize="15sp" />

```

```

    <TextView
        android:id="@+id/falseRow53"
        android:layout_marginStart="1dp"
        android:layout_weight="1"
        android:background="#FFFFFF"
        android:gravity="center"
        android:padding="5dp"
        android:text="@string/bigf"
        android:textColor="#000"
        android:textSize="15sp" />

```

```
</TableRow>
```

```
<TableRow>
```

```

    <TextView
        android:id="@+id/trueRow54"
        android:layout_height="match_parent"
        android:layout_marginStart="1dp"
        android:layout_weight="1"
        android:background="#FFFFFF"
        android:gravity="center"
        android:padding="5dp"
        android:text="@string/bigf"
        android:textColor="#000"
        android:textSize="15sp" />

```

```

    <TextView
        android:id="@+id/falseRow54"
        android:layout_marginStart="1dp"
        android:layout_weight="1"
        android:background="#FFFFFF"
        android:gravity="center"
        android:padding="5dp"
        android:text="@string/bigf"
        android:textColor="#000"
        android:textSize="15sp" />

```

```
</TableRow>
```

```
<TableRow>
```



```

        <TextView
            android:id="@+id/trueRow55"
            android:layout_height="match_parent"
            android:layout_marginStart="1dp"
            android:layout_weight="1"
            android:background="#FFFFFF"
            android:gravity="center"
            android:padding="5dp"
            android:text="@string/biggt"
            android:textColor="#000"
            android:textSize="15sp" />

        <TextView
            android:id="@+id/falseRow55"
            android:layout_marginStart="1dp"
            android:layout_weight="1"
            android:background="#FFFFFF"
            android:gravity="center"
            android:padding="5dp"
            android:text="@string/bigf"
            android:textColor="#000"
            android:textSize="15sp" />
    </TableRow>

    <TableRow>

        <TextView
            android:id="@+id/trueRow56"
            android:layout_height="match_parent"
            android:layout_marginStart="1dp"
            android:layout_weight="1"
            android:background="#FFFFFF"
            android:gravity="center"
            android:padding="5dp"
            android:text="@string/biggt"
            android:textColor="#000"
            android:textSize="15sp" />

        <TextView
            android:id="@+id/falseRow56"
            android:layout_marginStart="1dp"
            android:layout_weight="1"
            android:background="#FFFFFF"
            android:gravity="center"
            android:padding="5dp"
            android:text="@string/bigf"
            android:textColor="#000"
            android:textSize="15sp" />
    </TableRow>
</TableRow>
</TableLayout>

</LinearLayout>

<Space
    android:id="@+id/spaceAfterTable"
    android:layout_below="@+id/layForTable3"
    android:layout_width="match_parent"
    android:layout_height="25dp" />

<TextView
    android:id="@+id/explainForTable"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_below="@+id/spaceAfterTable"

```

```

        android:layout_centerInParent="true"
        android:layout_marginStart="20dp"
        android:text="@string/explainForTable"
        android:textColor="#000000"
        android:visibility="invisible" />

<Button
    android:id="@+id/nextStepExpl"
    android:layout_alignParentBottom="true"
    android:layout_marginBottom="30dp"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="@string/next"
    android:layout_centerHorizontal="true"
    android:visibility="invisible"/>

<ImageView
    android:id="@+id/animImage"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_centerHorizontal="true"
    android:layout_centerVertical="true"
    android:visibility="gone"/>

</RelativeLayout>

```